

SpamFilter ISP User Guide

© 2020 LogSat Software



SpamFilter ISP and SpamFilter Enterprise

Simple, effective, accurate, affordable anti-spam solutions

by LogSat Software

SpamFilter ISP is used by companies and ISPs running their own mail servers. SpamFilter uses dozens of different filters and techniques to detect and block spam.

A global network of SpamFilter ISP installations all cooperate to report spam signatures to our SpamFilter Distributed Database. Our SFDB, along with DNS-RBL based black lists, statistical Bayesian DNA fingerprinting, image scanning, more than a dozens of blacklist/white rules, country filters, SPF and SURBL filters, remote DNS validation tests, all contribute in providing unmatched protection against spam.

SpamFilter ISP User Guide

© 2020 LogSat Software

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: July 2020

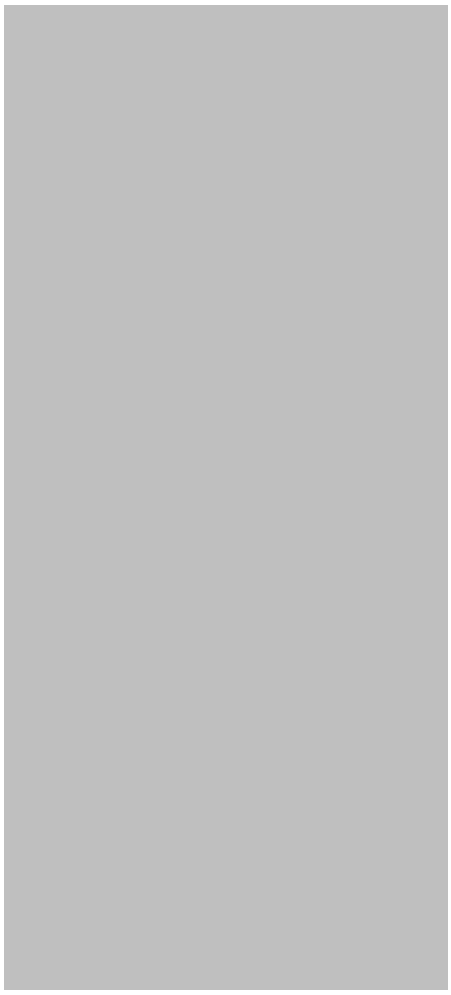


Table of Contents

Foreword 0

Part I Introduction 6

Part II Quick Setup 7

Part III SpamFilter Standard vs Enterprise - Feature Comparison 9

Part IV How it works 10

1 What emails can be blocked (blacklisted) 10
2 What emails are allowed to be delivered (whitelisted) 13

Part V SpamFilter Settings 15

1 Configuration tab 15
Filter Settings 17
2 Black / White Lists tab 19
Blacklists 19
Whitelists 22
Authorized TO Web Service 23
Regular Expressions 24
3 SFDB Filter - SpamFilter Distributed Blacklist 25
4 SFDC Filter - SpamFilter Distributed Content 26
5 GreyListing 26
6 SPF - Sender Policy Framework 27
7 Bayesian Statistical Filtering 30
8 SMTP Authentication 32
9 Filters per-Domain 32
10 Customized Items 32
11 Filter Order 34

Part VI Quarantine Database 34

1 SpamFilter ISP 35
2 SpamFilter Enterprise 38
3 Microsoft SQL Server Database Setup 40
4 MySQL Database Setup 42

Part VII Network Configurations Samples 44

1 SMTP servers directly connected to Internet 44
2 SMTP server with single IP address connected to Internet 45

© 2020 LogSat Software

3 SMTP servers behind Firewall 46
4 Single SMTP multihomed server behind Firewall 47
5 Single SMTP server behind Firewall 48

Part VIII Running SpamFilter 49

1 SpamFilter Service / Console Application 49
2 Running multiple SpamFilters 49

Part IX Upgrading SpamFilter ISP to SpamFilter Enterprise 51

1 Upgrading to SpamFilter Enteprise "offline" 51

Part X Antivirus Plugin 53

Part XI Log Analysis & Statistics 55

Part XII Web Interface for End-User Quarantine Access 55

1 Web Server Configuration 56
2 Microsoft Outlook Configuration 57

Part XIII SSL Certificates 58

1 Creating and self-signing your own certificate 59
2 Exporting an existing commercial SSL certificate 59

Part XIV Regular Expressions (RegEx) 61

Part XV SpamFilter.ini additional settings 68

Part XVI Purchase 77

Index 78

© 2020 LogSat Software

1 Introduction

Simple, effective, accurate, affordable anti-spam solutions

SpamFilter ISP is used by companies and ISPs running their own mail servers. SpamFilter uses dozens of different filters and techniques to detect and block spam.

A global network of SpamFilter ISP installations all cooperate to report spam signatures to our SpamFilter Distributed Databases. Our SFDB and SFDC, Greylisting, along with DNS-RBL based black lists, statistical Bayesian DNA fingerprinting, image scanning, more than a dozens of blacklist/white rules, country filters, SPF and SURBL filters, remote DNS validation tests, all contribute in providing unmatched protection against spam.

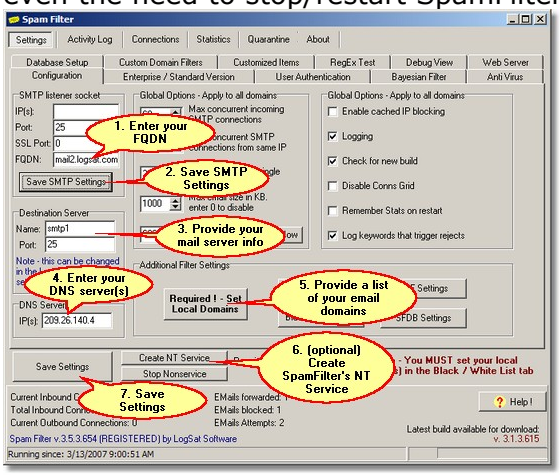
This guide is available in PDF, .CHM, and HTML formats, and is located in the \SpamFilter\Documentation directory.

© 2020 LogSat Software

2 Quick Setup

SpamFilter ISP can be installed and configured in just a few minutes. Server reboots are never required.

Following are the only steps required to implement SpamFilter on your network. Database functionality to quarantine spam emails can be added at a later time, without even the need to stop/restart SpamFilter.



1. **Install SpamFilter on a server. This can be done either by running the installer in the ZIP distribution file, or by simply extracting the** contents of the Win32 directory in the ZIP to a destination of your choice.
2. **Start SpamFilter**
   2.1. If you simply extracted the distribution ZIP file manually, execute the main program file "SpamFilter.exe".
   2.2. If you installed SpamFilter using the installer, run "SpamFilter ISP - Standalone mode" from the Start Menu. If you opted to have the installer create the SpamFilter NT service, you can optionally start SpamFilter now by starting the SpamFilter Service. Please see notes under SpamFilter Service / Console Application when running in service mode.
3. **In SpamFilter's Settings tab, perform the steps indicated in the screenshot above**.
   3.1. Enter your FQDN (Fully Qualified Domain Name). This is the server name that is sent in the SMTP welcome banner.
   3.2. Click on the "Save SMTP Settings" button to save the FQDN. Only the IP, ports and FQDN settings need to be saved using this button.
   3.3. Enter the name (or IP address) and port number of your default mail server, where SpamFilter will forward clean emails to. If you need to enter different mail servers for some or all of your domains, you can to that in step 3.5.
   3.4. Enter the IP(s) of your DNS server(s), separating multiple entries with commas.
   3.5. Provide a list of your local domains. SpamFilter will only accept emails if they are addressed to these domains.
   3.6. Click on "Save Settings"
4. **Configure your network so that incoming internet email is routed to SpamFilter. SpamFilter should be the first server/application that processes incoming emails.**

© 2020 LogSat Software

© 2020 LogSat Software

3 SpamFilter Standard vs Enterprise - Feature Comparison

SpamFilter ISP and SpamFilter Enterprise feature the same filtering capabilities, accuracy and performance. There are however differences in how the products can be customized to handle multiple email domains, and in how the filter settings are stored.

SpamFilter Enterprise has all the features of SpamFilter ISP. In addition, SpamFilter Enterprise allows the customizations of most filtering rules independently for each email domain being managed. This allows to fine-tune filtering rules for each of the email domains being hosted. If a server running SpamFilter Enterprise is handling emails for a hospital and for a law-firm, very different filtering rules can now be applied to each of the two domains.

SpamFilter Enterprise stores most of the filtering settings in a database. This allows companies running SpamFilter Enterprise to be able to more easily customize their installation of SpamFilter, adding functionality that may not be provided out-of-the-box. Multiple installations of SpamFilter Enterprise can all use and share the same database, allowing a centralized configuration for all your installations. Each SpamFilter Enterprise caches a copy of all database settings to a local hard drive, allowing SpamFilter to function even if the database server is offline or is not functioning.

	SpamFilter ISP	SpamFilter Enterprise
SFDB - SpamFilter Distributed Blacklist	✓	✓
MAPS DNS RBL Filters	✓	✓
SURBL Filters	✓	✓
SPF - Sender Policy Framework	✓	✓
Bayesian Statistical Filtering	✓	✓
Image Scanning / Filtering	✓	✓
Blacklist IPs	✓	✓
Blacklist Domains	✓	✓
Blacklisted FROM Emails	✓	✓
Blacklisted TO Emails	✓	✓
Blacklist by Country	✓	✓
Honeypot Capabilities	✓	✓
Attachment Blocking	✓	✓
Keywords Filter	✓	✓
Reverse DNS validation	✓	✓
MX Record validation	✓	✓
Reject if "Mail From" = "Mail To"	✓	✓
Reject if "From Domain"	✓	✓

© 2020 LogSat Software

	SpamFilter ISP	SpamFilter Enterprise
= "To Domain"		
Whitelist Domains / IPs	✓	✓
Deliver specific emails without filtering	✓	✓
Whitelist FROM Emails	✓	✓
List of "Authorized TO Emails"	✓	✓
SMTP User Authentication with SSL support	✓	✓
Customize filter settings for each domain		✓
Store settings in database		✓

4 How it works

Basic operation for SpamFilter ISP is as follows. SpamFilter is configured to handle your primary MX record for incoming emails (see DNS details here). SpamFilter can be configured to listen on a specific IP, multiple IPs or all IPs bound to the NIC card. See Configuration Section for more details.

4.1 What emails can be blocked (blacklisted)

SpamFilter ISP's strength lies in the large number of filters that are applied to detect spam. Below is a partial list of some of the various filters that are employed to stop spam.

SFDB - Spam Filter Distributed Blacklist Database
The SFDB filter has been developed exclusively by LogSat Software. This filter uses a very powerful resource to stop spam: The entire global SpamFilter ISP user community.

This latest filter is proving to be one of the most effective and accurate tools in stopping spam.
Anytime a company running SpamFilter ISP blocks an email, the sender's IP address is sent to our centralized SFDB database. This allows the SFDB filter to have access to a huge repository of spammer's IPs, updated in real-time by all the SpamFilter ISP installations in the world.
Our database analyzes this data in realtime, and will block IPs that have sent excessive amounts of spam to multiple locations in the world in the spam of the previous few minutes. This allows the SFDB to be extremely accurate, effective, and to start blacklisting IPs within minutes of them sending spam.
IP addresses from the database are automatically aged and removed from the database

© 2020 LogSat Software

within 6-24 hours if they stop sending spam and/or viruses.

SFDC - SpamFilter Distributed Content Database
The SFDC (SpamFilter Distributed Content) filter is the latest filtering technology developed by LogSat Software.

When SpamFilter ISP receives an email, it will analyze the email's contents and will calculate a 20-byte hash to characterize it. We developed technology that is able to detect similar emails based on their contents. SpamFilter will assign the same hash to similar emails. When SpamFilter detects that emails with the same hash signature are originating from several different locations, it will report such anomaly to our centralized servers.

Our database analyzes, in real-time, this incoming flow of messages, and, based on their quantity, origin and destinations, is able to detect what signature hashes are generated by spam emails.

The technology behind the SFDC allows our centralized database to detect spam signatures regardless of the email's text and contents, but rather base it on the patterns used by spammers to deliver their emails.

Detection of spam signatures in images
SpamFilter ISP contains proprietary technology developed by LogSat Software that scans images embedded in emails for spam content.

We at LogSat Software were the first, in June 2007, to develop technology that allowed SpamFilter to scan images embedded in PDF files for spam content (the so-called PDF spams).

RBL and SURBL Blacklists
Spam Filter ISP can check any user-specified RBL blacklist to see if the sender's IP address is being blacklisted. Reliability can be improved by requiring an IP to be blacklisted by two or more RBL servers for it to be marked as spam.

Spam Filter will analyze all URLs specified in the email body itself, and will check any user-specified SURBL blacklist server to see if the URL in the email is being used to host spam-related websites.

Greylisting
Greylisting is not an anti-spam filter itself. More specifically, greylisting takes advantage of a required behavior by the RFCs that some anti-spam products use to greatly reduce the amount of spam received.

In the majority of the cases, when a "spam bot" computer is used to send spam, it will do so by sending huge amounts of emails in the fastest way possible. If a recipient's SMTP server does not respond, chances are that the spam bot will ignore such server and move on.

Luckily this behavior by spammers is in direct violation of the RFCs that dictate how email works. The RFCs require that, if an initial attempt to deliver an email fails, the sender must retry to send it.

Greylisting takes advantage of this by initially denying every connection attempt from

© 2020 LogSat Software

an IP address. Only after a certain, small amount of time is the remote IP allowed to connect. If the sender is a spam bot, it is very likely that said IP will never retry to connect again, and so it will not even try to send spam. If the sender is a legitimate server, they will be following the RFC guidelines, and within a few minutes they will retry sending the email, which will be then delivered.

SpamFilter ISP v4 and higher support greylisting, and we at LogSat Software have made some changes in the implementation of this method to reduce the amount of delays that occur when a server connects for the first time to SpamFilter.

Bayesian statistical DNA fingerprinting
Spam Filter ISP performs statistical DNA fingerprinting on all incoming emails. This bayesian filter is self-learning, continuously analyzing your incoming traffic to improve its accuracy with time.

SSL and SMTP Authentication
Many mail servers lack support for SSL and SMTP Authentication. SpamFilter ISP supports both SSL and SMTP AUTH via Active Directory, LDAP, and Unix-style password files. If a user is authenticated, they will be able to bypass all filtering rules and use SpamFilter ISP as a relay to send their outgoing emails.

Administrators can then add support for SMTP Authentication (and SSL) if they have older mail servers that do not have these features.

SPF - Sender Policy Framework
SPF fights email address forgery and makes it easier to identify spam, worms, and viruses. SPF is an open source standard that is emerging as a solution to prevent spammers from using fake email addresses. Domain owners identify sending mail servers in DNS. SpamFilter ISP verifies the envelope sender address against this information, and can distinguish legitimate mail from spam before any message data is transmitted
Block Emails from User-Defined Countries

SpamFilter ISP is able to block emails being sent from any user-specified country. In addition, SpamFilter will track and record the number of email attempts made from all countries. This allows administrators to determine, visually, if there are any countries they do not wish to receive emails from.

...and dozens of more filters!
In addition to the filter specified above, SpamFilter ISP supports several more filters that can be used to detect spam.

A partial list is below.

- Local IP Blacklist - Our SPAM Filter server checks if the remote server's IP address matches an entry in your local IP blacklist file, the email is rejected.
- Local Domain Blacklist - The SPAM Filter gateway checks if the domain portion in the sender's email address is in your local domain blacklist file, the email is also rejected.
- Local FROM EMail Blacklist - The sender's email address is checked against your local list of blacklisted email addresses. If present, it is rejected.
- Local TO EMail Blacklist - The recipient's email address is checked against your local list of blacklisted email addresses. If present, it is rejected.
- Attachment Blocking - SPAM Filter can check emails for specific attachments or

© 2020 LogSat Software

attachment extensions. If found, the email is rejected.
- Keyword Content Filtering - Our SPAM Filtering software can check email content and subject for specific keyword and/or phrases. If found, the email is rejected.
- Honeypot Emails - You can have a list of "honeypot" email addresses. Any email sent to an address in the list will cause the sender's IP to be blacklisted.
- Connections can be rejected if the remote server does not have a reverse DNS PTR entry.
- Spam Filter is able to check if the sender's MX DNS record is valid before accepting email.
- Refuse connections if there are too many spaces in the subject line.
- Max Recipients in single session - Use this setting to limit how many RCPT TO commands can be issued in a single session.
- Max Email Size - Incoming emails can be blocked if they exceed a certain size.
- Reject if Empty "Mail From" - If this option is checked SPAMFilter will reject all emails with an empty "Mail From" field.
- Reject if "Mail From" = "Mail To" - Reject all emails where the sender's email is the same as the recipient's email.
- Reject if "From Domain" = "To Domain" - SPAM Filter can reject all email where the sender's domain is the same as the recipient's domain.
- Tag Spam & Deliver - Allows to tag spam by adding the header "X-SF-SPAM:Y" to email classified as spam. The email is then forwarded to the destination SMTP server. This allows administrators to handle spam as they wish on the back-end.
- Tag Spam in Subject & Deliver - Allows to tag spam by prefixing the word SPAM: in the subject line of emails classified as spam. The email is then forwarded to the destination SMTP server. This allows administrators to handle spam as they wish on the back-end.

4.2 What emails are allowed to be delivered (whitelisted)

- **Allowed domains -** If the IP passes the DNS tests SpamFilter then checks the recipient domain. If the domain is listed as a local domain, then the recipient is accepted. This is done to prevent spammers to use SpamFilter to relay. **It is very important to add your own domains to the local domain list. If not, you will not be able to receive email.**
- **Excluded IPs -** If an IP is blacklisted, but you *really* need to be able to receive email from that domain anyways, the domain can be added in an exclude list as to allow it to bypass the blacklist rules.
- **Excluded Domains -** If an IP is blacklisted but you still wish to receive email from them, the IP can be added to an IP exclude list to allow it to bypass the blacklist rules.
- **Unfiltered Emails -** If you have users who do not want to receive filtered emails, they can be accommodated by adding them to a pass-list. Emails addressed to them will bypass all of SpamFilter rules.
- **Excluded FROM Emails** - If you want a sender's email address to be excluded from all filtering rules, you can add it to an exclude list.
- **Authorized TO Emails** - If you want SpamFilter to *only* deliver emails to specific addresses in your domain(s), you can manage such a list here. Please not that if such a list is present, SpamFilter will not deliver email to an address unless it is present in such a list. Use with care.

© 2020 LogSat Software

- **Keyword whitelisting -** You can provide your customers with specific keywords that, if found in the body or subject of emails, will bypass all filtering rules.
- **User Authentication** - SpamFilter supports sender's authentication via Active Directory, LDAP, and Unix-style password files.

© 2020 LogSat Software

5 SpamFilter Settings

5.1 Configuration tab

The following global options are available. They apply to all emails processed by SpamFilter:

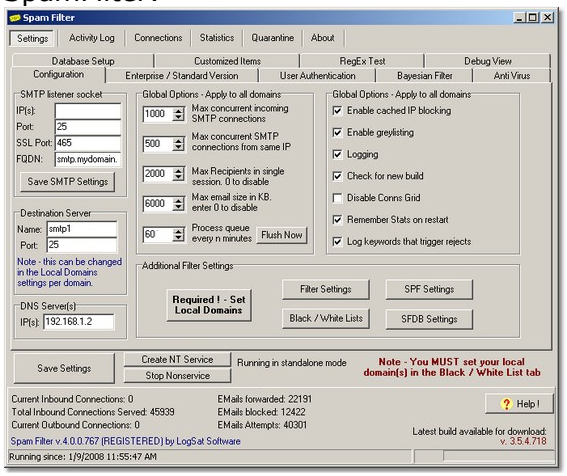


Figure 1

- **SMTP listener socket**
  - **IP(s) -** SpamFilter can be configured to listen on a specific IP, multiple IPs (separate the IPs with a comma in the IP input box), or all IPs (leave the IP field blank).
  - **Port -** The port on which SpamFilter will listen on. 25 is the standard SMTP port.
  - **SSL Port -** SpamFilter supports SSL with SMTP. The standard SSL port is 465, however SSL is disabled by default by setting the port to 0. Please see the SSL certificates section for more information about using your own certificates.
  - **FQDN -** This is the Fully Qualified Domain Name that SpamFilter will output in the welcome banner sent to initiating connections.
  - Destination Server
  - **Name -** The name or IP address of the default server where your email will be forwarded to. Please note that this default value can be overridden for individual domains if you need to have multiple routes for your domains. Please see the section on Local Domains for more info.
  - **Port -** The port on which email will be forwarded to. Having a port other that 25 can be useful in situations where you only have access to a single IP. If you configure your destination MTA server to listen on a port other than 25, you can have both SpamFilter and your MTA co-exist on a single IP address.
    Note - if the destination server is unavailable, emails are saved to the *queue* directory. Redelivery of items in the queue is attempted every 60 minutes, and also every time SpamFilter is started.
- **DNS Server**
  - **IP(S) -** Enter the IP of your DNS server here. You can enter multiple DNS servers for redundancy by separating them with commas.
- **Global Options**
  - **Max concurrent incoming SMTP connections -** You can limit the maximum number of concurrent incoming connections here.
  - **Max concurrent incoming SMTP connections from single IP-** You can further

© 2020 LogSat Software

limit the maximum number of concurrent connections originating from an single IP address with this setting.

- **Max Recipients in single session -** Use this setting to limit how many RCPT TO commands can be issued in a single session. Enter a value of "0" to disable this setting.
- **Max Email Size** - Incoming emails can be blocked if they exceed a certain size. Enter a "0" to disable this setting.
- **Process queue every n minutes -** Use this setting to control how often SpamFilter attempts to redeliver the items on hold in the queue directory.
- **Enable Cached IP Blocking -** If an IP address sends more than a certain number of spam emails (3 by default) during a certain time interval (10 minutes by default), then it can be temporarily banned (blacklisted). All further connections from that IP address will be immediately rejected without allowing the sender to transmit any data. This should greatly reduce the load on the server. A banned IP address will be automatically removed from this temporary blacklist after a defined time interval (60 minutes by default). To prevent specific IPs to be added to this list, they can be added to DoNotAddIPToHoneypot SpamFilter.ini option.
- **Enable GreyListing Filtering** - Greylisting initially denies every connection attempt from an IP address. Only after a certain, small amount of time is the remote IP allowed to connect. If the sender is a spam bot, it is very likely that said IP will never retry to connect again, and so it will not even try to send spam. If the sender is a legitimate server, they will be following the RFC guidelines, and within a few minutes they will retry sending the email, which will be then delivered.
- **Logging -** Check this box to enable logging in the log directory.
- **Auto-check for new build -** If checked SpamFilter will connect with our website to see if a new version of the software is available. SpamFilter will issue a simple GET request to http://logsat.com/spamfilter/version.htm to retrieve the current version number.
- **Disable Connections Grid -** The Connections tab will show you in real-time what the various connections on your servers are and what they are doing. If you have a busy site with 500 concurrent connections this list can get pretty crowded and could affect performance. This feature can thus be disabled from here.
- **Remember Stats -** Check this box to save the email statistics when shutting down SpamFilter.

© 2020 LogSat Software

5.1.1 Filter Settings

The following settings are available. In SpamFilter Enterprise, each email domain can be configured independently with different settings. The "ALL DOMAINS" is a reserved entry that contains default filtering values to all domains. Each domain can be given customized filter settings by clearing the "Use Defaults" checkbox.

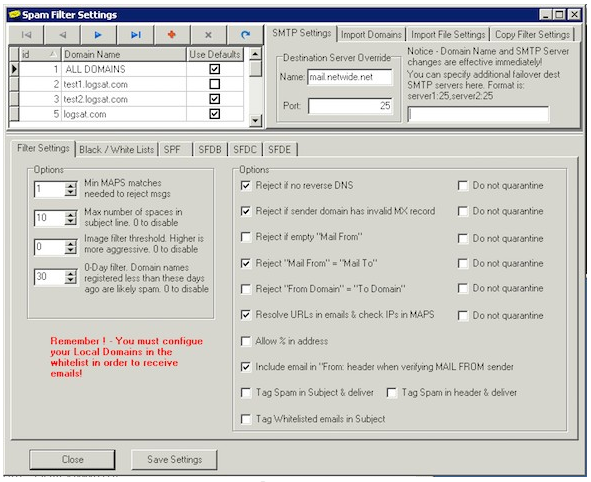


Figure 2

- **Min MAPS matches needed to reject msgs -** Sometimes MAPS blacklists can be too strict and list legitimate domains in their blacklists. You can reduce the number of false positive by requiring that more than one single blacklist match is found before rejecting a connection.
- **Max number of spaces in subject line -** Many spam messages contain a large number of spaces and tabs, and SpamFilter can reject emails based on this setting. Enter a "0" to disable this filter.
- **Image filter threshold –** SpamFilter scans *inside* images embedded in emails to detect spam signatures. Use this setting to adjusts the threshold of the image filter. The higher the threshold, the more aggressive the filter. Enter a "0" to disable the image filter.
- **0-Day filter** - Many spammers register bogus domains for the sole purpose of sending spam, and use them right away. Legitimate domains usually take longer to purchase, configure, test and implement the new domain name. SpamFilter can reject emails if the domain is recently registered (by default less than 30 days before the email is received).
- **Reject if no reverse DNS -** SpamFilter can be configured to reject emails if the remote server does not have a valid reverse DNS PTR entry.
- **Reject if sender domain has invalid MX record –** SpamFilter can check to see if the sender's email domain has a valid MX record. If the MX record is not present in the DNS, the email will be rejected. The email domain in this case is what follows the "@" sign in the sender's email address. For example, is the sender is user@ *sub.domain.com*, the domain that will be tested for the presence of an MX record is *sub.domain.com*.
- **Reject if Empty "Mail From" -** If this option is checked SpamFilter will reject all emails with an empty "Mail From" field. Please note that this setting will delete

© 2020 LogSat Software

legitimate email, as in email receipt notifications and some error emails.

- **Reject if "Mail From" = "Mail To" -** Reject all emails where the sender's email is the same as the recipient's email. Note that this causes problems with users who send emails to themselves using EBay's web interface for example.
- **Reject if "From Domain" = "To Domain" -** SpamFilter can reject all email where the sender's domain is the same as the recipient's domain. Usually your users will not go thru SpamFilter when sending emails to themselves, Spammers often use this technique
- **Resolve URLs in emails & check IP in MAPS -** SpamFilter can optionally resolve all URLs embedded in an email message to IP addresses, and then check those IP addresses in the MAPS servers to see if any of them are being blacklisted.
- **Allow % in address -** SpamFilter can then optionally check to see if the recipient address has a % sign in it. Many SMTP servers are susceptible to being tricked into relaying mail with this. Ex. if you are isp.com, then a spammer could try to use joe%yahoo.com@isp.com to relay mail to joe@yahoo.com if your server is vulnerable.
- **Include email in "From:" header when verifying MAIL FROM sender -** Use this option to include the email address in the "From:" header, in addition to the real envelope email address specified in the MAIL FROM command, when checking the sender's email agains the "FROM Emails" blacklist, the "Domains" blacklist, the "Excluded FROM Emails" whitelist, and the "Excluded Domains / IPs" whitelist.
- **Tag Spam & Deliver -** Allows to tag spam by adding the header *"X-SF-SPAM:Y"* to email classified as spam. The email is then forwarded to the destination SMTP server. This allows administrators to handle spam as they wish on the back-end.
- **Tag Spam in Subject & Deliver -** Allows to tag spam by prefixing the word SPAM: in the subject line of emails classified as spam. The email is then forwarded to the destination SMTP server. This allows administrators to handle spam as they wish on the back-end.
- **Tag Whitelisted emails in Subject -** When checked, this option causes SpamFilter to add a tag in the subject line (customizable via the "WhitelistTagPrefix" parameter in the SpamFilter.ini file) to indicate that the email was whitelisted.

© 2020 LogSat Software

5.2 Black / White Lists tab

The following blacklists and whitelist options are available. In SpamFilter Enterprise, each setting can be configured independently for each email domain.

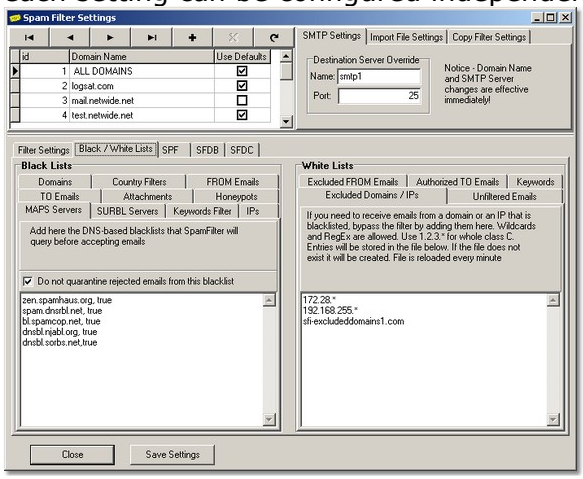


Figure 3

5.2.1 Blacklists

- **MAPS Blacklist servers -** SpamFilter checks the IP address initiating the connection. If it is listed in one of its many DNS blacklists the connection is refused. SpamFilter can reject connections based on a configurable minimum number of matches. A "**,true**" after an RBL entry means their DNS is expecting the IP to be reversed, i.e. to test a connection from 1.2.3.4 they expect 4.3.2.1.bl.spamcop.net
- **SURBL Blacklist servers -** SpamFilter scans the content of emails for any HTTP links and URLs. Every link found is then tested against one of the many SURBL DNS blacklists available. If present, the connection is refused.
- **Blacklisted IPs -** You can keep a file with additional IPs that you want to blacklist by entering the filename below. If the file does not exist it will be created. The file is reloaded every minute. List individual IP addresses on each line. Use an ending .0 for a Class C wildcard (i.e. 192.12.45.0 to block 192.12.45.1 --> 192.12.45.255, or 192.12.0.0 to block 192.12.0.0 --> 192.12.255.255). This IP blacklist also supports the use of CDIR notation to specify networks. For example, 192.12.45.0/24 will block the previous Class C of addresses as well. The contents of the file will be loaded in the memo box, allowing you to make changes to the file.
- **Blacklisted Domains -** You can keep a file with additional Domains that you want to blacklist (based on the MAIL FROM field) by entering them below. Enter one domain per line, wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx). If the file does not exist it will be created. The file is reloaded every minute. The contents of the file will be loaded in the memo box, allowing you to make changes to the file.
  - This list supports the **:NULL** option to send emails in a black hole. If an entry is in the form ***domain1.com:NULL*** it will cause all emails from domain1.com to be accepted and then sent to NULL right away. Such emails will not cause NDRs, they will not be quarantined, they will not be seen by the users.
  - If an entry is in the form ***domain1.com:NoNDR*** such emails will not cause NDRs as in the DoNotSendNDROnQuarantine parameter in the ini file.
  - This list supports the **:Honeypot** option, which will cause the sender's IP address to

© 2020 LogSat Software

be automatically blacklisted in the future.
  - This list supports the **:ForceDelete** option, which will prevent the email from being archived in the quarantine database if stopped by this blacklist.
- **Blacklisted FROM Emails -** If you want to block any particular email addresses, enter them here, one email per line. Wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx).
  - This list supports the **:NULL** option to send emails in a black hole. If an entry is in the form ***user1@domain1.com:NULL*** it will cause all emails from user1@domain1.com to be accepted and then sent to NULL right away. Such emails will not cause NDRs, they will not be quarantined, they will not be seen by the users.
  - If an entry is in the form ***domain1.com:NoNDR*** such emails will not cause NDRs as in the DoNotSendNDROnQuarantine parameter in the ini file.
  - This list supports the **:Honeypot** option, which will cause the sender's IP address to be automatically blacklisted in the future.
  - This list supports the **:ForceDelete** option, which will prevent the email from being archived in the quarantine database if stopped by this blacklist.
- **Blacklisted TO Emails -** If you want to block any particular destination addresses, enter them here, one email per line. Wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx).
  - If an entry is in the form ***user1@domain1.com:NULL*** it will cause all emails to user1@domain1.com to be accepted and then sent to NULL right away. Such emails will not cause NDRs, they will not be quarantined, they will not be seen by the users.
  - If an entry is in the form ***domain1.com:NoNDR*** such emails will not cause NDRs as in the DoNotSendNDROnQuarantine parameter in the ini file.
  - This list supports the **:Honeypot** option, which will cause the sender's IP address to be automatically blacklisted in the future.
  - This list supports the **:ForceDelete** option, which will prevent the email from being archived in the quarantine database if stopped by this blacklist.
- **Country Filters -** SpamFilter checks the what country incoming connections are coming from. The current number of connections for each country can be updated by clicking on the Update Stats Now button. Columns can be sorted by clicking on the column header. This will help you in sorting countries and hits so you can determine if there are any countries you do not wish to receive email from.
- **Honeypot Emails -** You can have a list of "honeypot" email addresses. Any email sent to an address in the list will cause the sender's IP to be blacklisted. The IP address will be added to the file *HoneypotBlockedIPs.txt*, which contains the list of blocked IPs automatically added by this filter. This filter is typically used by adding non-existent email accounts to it that you know should never receive mail. If they do, then the email is likely spam, so the remote IP will be blacklisted automatically.
- **Attachment Blocking -** You can block emails that have unwanted attachments. You can keep a file with banned attachments here. check emails for specific attachments or attachment extensions. If the attachment is found, the email is rejected. Wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx).
  - This list supports the **:NULL** option to send emails in a black hole. If an entry is in the form ***filename:NULL*** it will cause all emails with the filename attachment to be accepted and then sent to NULL right away. Such emails will not cause NDRs, they will not be quarantined, they will not be seen by the users.
  - If an entry is in the form ***domain1.com:NoNDR*** such emails will not cause NDRs as in the DoNotSendNDROnQuarantine parameter in the ini file.

© 2020 LogSat Software

- This list supports the **:Honeypot** option, which will cause the sender's IP address to be automatically blacklisted in the future.
- This list supports the **:ForceDelete** option, which will prevent the email from being archived in the quarantine database if stopped by this blacklist.

- **Keywords Filter -** You can check email content and subject header for specific keyword and/or phrases. If found, the email is rejected. You can also use Regular Expressions (RegEx). If the keyword file does not exist it will be created. The file is reloaded every minute. The contents of the file will be loaded in the memo box, allowing you to make changes to the file.
  - This list supports the **::NULL** option to send emails in a black hole.
  - If an entry is in the form ***keyword::NULL*** it will cause all emails to be accepted and then sent to NULL right away. Such emails will not cause NDRs, they will not be quarantined, they will not be seen by the users.
  - If an entry is in the form ***keyword::NoNDR*** such emails will not cause NDRs as in the DoNotSendNDROnQuarantine parameter in the ini file.
  - This list supports the **::Honeypot** option, which will cause the sender's IP address to be automatically blacklisted in the future.
  - This list supports the **:ForceDelete** option, which will prevent the email from being archived in the quarantine database if stopped by this blacklist.
  - This list supports the **::NEGATE** suffix in blacklist keywords. Adding this suffix causes a MATCH if the keyword is NOT present, or causes a MISMATCH if the keyword IS present.

  Please note that unlike in other cases, with the keyword list you must enter the ":" symbol *twice* to specify the extra tag.
  The keyword list allows to specify multiple RegEx expressions separated by commas, just as regular keywords can be separated by commas. This has the effect of specifying "AND" rules for RegEx. Note that a "Standard non-RegEx keyword must be specified first for SpamFilter to recognize this syntax. For example: X-SF,([a-z]), ([0-9])
  The keyword rules are as follows:
  - Keyword(s) are entered on separate lines
  - If all keyword(s) on a single line are found in the message, it is then rejected.
  - Separate keywords on a single line by using commas.
  - The subject of an email can be scanned for keywords by prefixing the keyword list with **Subject:**
  - The contents of the "From:" header of an email (which usually includes a name and an email address) can be scanned for keywords by prefixing the keyword list with **EmailFrom:**
  - The contents of the "To:" header of an email (which usually includes one or more names and email addresses) can be scanned for keywords by prefixing the keyword list with **EmailTo:**
  - The complete set of headers of an email can be scanned for keywords by prefixing the keyword list with **Headers:**

Sample keyword entries:	Sample email content and effects:

© 2020 LogSat Software

▪ mortgage,click here,mailing ▪ free,mailing,list ▪ unsubscribe ▪ Subject:viagra	.... low **mortgage**, **click here** to be removed from our **mailing** ...	rejected	matches all keywords in 1st line
	.... low **mortgage**, *click* over *here* to be removed from our **mailing** ...	accepted	click over here is no match for **click here**
	.... low mortgage, *click* over *here* to **unsubscribe** from our mailing ...	rejected	matches single keyword on 3rd line

5.2.2 Whitelists

- **Local Domains -** SpamFilter will only deliver email to the domains listed here. Wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx). If the domain in the RCPT TO email address is listed as a local domain, then the recipient is accepted. This is done to prevent spammers to use SpamFilter to relay email to third party email addresses/servers. **It is very important to add your own domains to the local domain list. If not, you will not be able to receive email.** If you need to have any domain listed here forward its destination email to a different server than the default destination server, you can specify so here. You can override the default destination server by appending a different mail server and port to any domain in this list. The syntax should be as follows (additional destination servers for redundancy are supported):
  DomainName:DestinationServer:DestinationPort[,DestinationServer:DestinationPort]
  For example:
  logsat.com:mail.logsat.com:25
  or
  logsat.com:mail.logsat.com:25,mail2.logsat.com:26

- **Excluded Domains / IPs -** You can keep a file containing a list of any "MAIL FROM" domains or any IPs from which you want to receive email if they would be blocked by any of your blacklist rules. Enter as many IPs or domains as you wish, one per line. Wildcards (* and ?, same rules as DOS wildcards) are allowed. To exclude a whole class C, enter it as 209.20.21.***.** If the file does not exist it will be created. The file is reloaded every minute.
- **Unfiltered Emails -** Any local email address listed here will cause SpamFilter to bypass all blacklist rules for it. If you have any users who do not want to have their email filtered, enter them here. Wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx).This list supports two :TAG options to bypass the default "pass all" rule for entries on this list.
  - If an entry is in the form ***user@domain1.com:TAGSUBJECT*** it will cause all emails sent to user@domain1.com to be accepted and then delivered to that user no matter what. However emails that are classified as spam by the various filters will have the prefix "SPAM:" added to the subject line.
  - If an entry is in the form ***user@domain1.com:TAG*** it will cause all emails sent to user@domain1.com to be accepted and then delivered to that user no matter what. However emails that are classified as spam by the various filters will have the header *"X-SF-SPAM:Y"* added to them.

© 2020 LogSat Software

- **Excluded FROM Emails** - You can keep a file containing a list of sender's email address to be excluded from all filtering rules. Enter one email address per line, wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx). If the file does not exist it will be created. The file is reloaded every minute. The contents of the file will be loaded in the memo box, allowing you to make changes to it.
- **Authorized TO Emails** - You can keep a file containing a list of authorized recipients. If you want SpamFilter to *only* deliver emails to specific addresses in your domain(s), you can manage such a list here. Enter one email address per line, wildcards (* and ?, same rules as DOS wildcards) are allowed. You can also use Regular Expressions (RegEx). If the file does not exist it will be created. The file is reloaded every minute. Please not that if such a list is present, SpamFilter will not deliver email to an address unless it is present in such a list. Use with care. **Delete the filename from the edit box to disable the list.**
  In addition to the local list of valid recipient email addresses, SpamFilter supports the use of a custom webservice hosted in your environment to query for valid users. More details on how to configure the webservice can be found in the next page.
- **Keywords Filter -** You can check email content and subject header for specific keyword and/or phrases. If found, the email is allowed through the filters. Useful if you want to allow certain customers to send you email without having to place them all in a email address whitelist. The same syntax rules as the blacklist keywords apply.
- **SMTP User Authentication -** SpamFilter is able to authenticate senders via LDAP, Active Directory, and a Unix-style password file. If a user is successfully authenticated, they will be whitelisted and will be able to use SpamFilter to relay emails. This will allow users to use SpamFilter's server as their "Outgoing SMTP Server" in their email client configuration. SSL support is available to encrypt the login credentials. Please see the SSL certificates section for more information about using your own certificates.

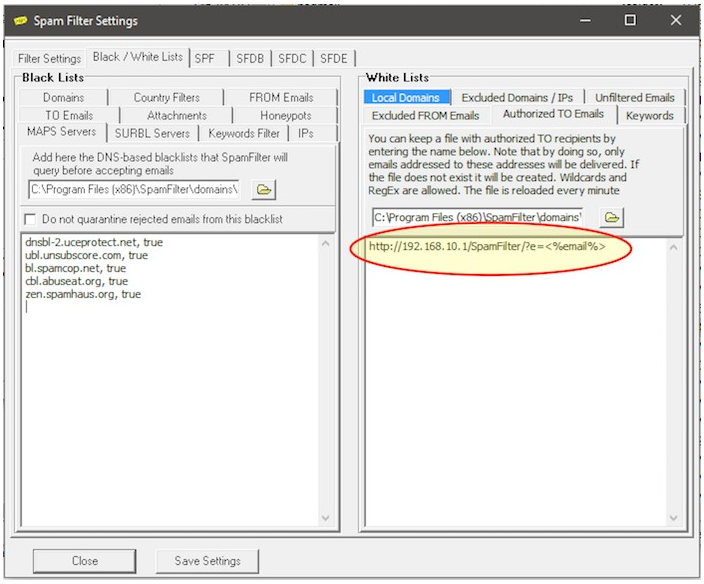
5.2.2.1 Authorized TO Web Service

In addition to supporting a local list of email addresses for the Authorized TO whitelist, SpamFilter can verify the validity of recipient email addresses by querying a webservice hosted in your environment. This is done by entering in SpamFilter the URL that SpamFilter needs to use to test for the validity of an email address using this format:
http://192.168.10.1/test/?e=*<%email%>*

SpamFilter will dynamically replace the <%email%> placeholder with the current email address being checked. You can use https if you'd like, and add other parameters - as long as you have one with <%email%> that will work. For example this will be fine too:
http://192.168.10.1/test/?e=*<%email%>*&user=bob

This is what the GUI looks like. For SpamFilter Enterprise you're able to specify a custom webservice URL for each domain you host:

© 2020 LogSat Software



You can use any directory you want in the URL, and any parameter name you prefer, for example:
http://192.168.10.1/test/?e=<%email%>
http://192.168.10.1/SpamFilter/?auth=<%email%>

When you create the webservice, the webserver must just reply with just a "NO" string if the user does not exist. With any other response, or with any type of error, SpamFilter will assume the address is instead valid.

So for example, when checking the non-existent email address "roberto333@logsat.com ", this would the the request being made to your webserver in the above example:

```
GET /test/?e=roberto333@logsat.com HTTP/1.1

```

And this is the webserver's response to indicate the user does not exist. Any other response where the body is different than a plain "NO" will be interpreted as a valid user.

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
Set-Cookie: ASPSESSIONIDACBDSDSR=ONJBANBANDJHHGMFNGPMBJKI; path=/
X-Powered-By: ASP.NET
Content-Type: text/html
Date: Thu, 21 Jan 2016 01:53:05 GMT
Content-Length: 2
Cache-Control: private
Connection: close

NO
```

5.2.3 Regular Expressions

Every single LINE of text in every single local list that is currently able to support wildcards will be treated with the usual wildcard rules. But if any one of those lines of text starts with an open parenthesis "(", and end with a closing parenthesis ")", then that one single line will be tested using RegEx rules. Please see the section for regular expression syntax for additional details.

© 2020 LogSat Software

5.3 SFDB Filter - SpamFilter Distributed Blacklist

The SFDB filter uses a very powerful resource to stop spam:
The entire global SpamFilter ISP user community.
Anytime an IP address is added to SpamFilter's local IP blacklist cache, the SFDB filter updates our Distributed Blacklist centralized database. This allows the SFDB filter to have access to a huge repository of spammer's IPs, updated in real-time by all the SpamFilter ISP users in the world. IP addresses from the database are automatically aged and removed from the database within 24 hours if they receive no further reports.
The SFDB filter detects spam by checking IP addresses against the SFDB database. The "network reliability" level tells SpamFilter how many different users must have reported a specific IP in order to classify it as spam.
Use a Network Reliability value of 0 to disable the SFDB filter.
In SpamFilter Enterprise, each setting can be configured independently for each email domain . In the free version of SpamFilter, the SFDB filter will only query the SFDB database, reporting of new IPs is disabled. Furthermore, in the free version, the SFDB filter is limited to a 15-day trial period.

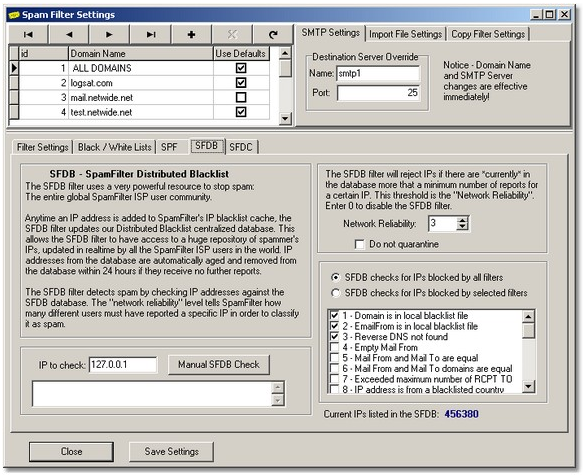


Figure 4

© 2020 LogSat Software

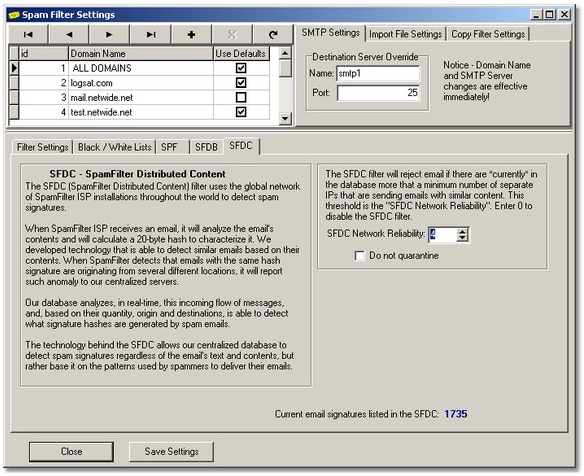
5.4 SFDC Filter - SpamFilter Distributed Content

The SFDC (SpamFilter Distributed Content) filter is the latest filtering technology developed by LogSat Software.

When SpamFilter ISP receives an email, it will analyze the email's contents and will calculate a 20-byte hash to characterize it. We developed technology that is able to detect similar emails based on their contents. SpamFilter will assign the same hash to similar emails. When SpamFilter detects that emails with the same hash signature are originating from several different locations, it will report such anomaly to our centralized servers.

Our database analyzes, in real-time, this incoming flow of messages, and, based on their quantity, origin and destinations, is able to detect what signature hashes are generated by spam emails.

The technology behind the SFDC allows our centralized database to detect spam signatures regardless of the email's text and contents, but rather base it on the patterns used by spammers to deliver their emails.



5.5 GreyListing

Greylisting is not an anti-spam filter itself. More specifically, greylisting takes advantage of a required behavior by the RFCs that some anti-spam products use to greatly reduce the amount of spam received.

In the majority of the cases, when a "spam bot" computer is used to send spam, it will do so by sending huge amounts of emails in the fastest way possible. If a recipient's SMTP server does not respond, chances are that the spam bot will ignore such server and move on.

Luckily this behavior by spammers is in direct violation of the RFCs that dictate how email works. The RFCs require that, if an initial attempt to deliver an email fails, the sender must retry to send it.

© 2020 LogSat Software

Greylisting takes advantage of this by initially denying every connection attempt from an IP address. Only after a certain, small amount of time is the remote IP allowed to connect. If the sender is a spam bot, it is very likely that said IP will never retry to connect again, and so it will not even try to send spam. If the sender is a legitimate server, they will be following the RFC guidelines, and within a few minutes they will retry sending the email, which will be then delivered.

SpamFilter ISP v4 and higher support greylisting, and we at LogSat Software have made some changes in the implementation of this method to reduce the amount of delays that occur when a server connects for the first time to SpamFilter.

Please note - Implementing greylisting will *initially* cause a slight delay when receiving emails. This delay will only occur *once* for each mail server that sends an email to SpamFilter. As soon as the mail server retries sending the email (which usually occurs after a few minutes), from then on all future emails from that server will be allowed to pass the greylisting filter. So if company A sends you an email, the very first attempt will be delayed by a few minutes. From then on, all emails will be delivered normally. This slight initial delay is seldom ever noticed by the end-users, as it occurs only once.

5.6 SPF - Sender Policy Framework

SPF is an open source standard that is emerging as a solution to prevent spammers from using fake email addresses. The following description was taken from the official SPF website at http://spf.pobox.com:
Domains use public records (DNS) to direct requests for different services (web, email, etc.) to the machines that perform those services. All domains already publish email (MX) records to tell the world what machines receive mail for the domain.
SPF works by domains publishing "reverse MX" records to tell the world what machines send mail from the domain. When receiving a message from a domain, the recipient can check those records to make sure mail is coming from where it should be coming from.
With SPF, those "reverse MX" records are easy to publish: one line in DNS is all it takes. Suppose a spammer forges a hotmail.com address and tries to spam you.
He connects from somewhere other than hotmail.
When his message is sent, you see MAIL FROM: <forged_address@hotmail.com>, but you don't have to take his word for it. You can ask Hotmail if the IP address comes from their network.
(In this example) Hotmail publishes an SPF record. That record tells you (your computer) how to find out if the sending machine is allowed to send mail from Hotmail.
If Hotmail says they recognize the sending machine, it passes, and you can assume the sender is who they say they are. If the message fails SPF tests, it's a forgery. That's how you can tell it's probably a spammer.
SpamFilter ISP looks up SPF DNS records for all incoming emails. If an SPF record exists, the query results can be any one of the following:

- Pass: the message meets the domain's definition of legitimacy.
- Neutral : the message does not meet a domain's definition of legitimacy, but the SPF client MUST proceed as if a domain did not publish SPF data. Likely used by domains in transition phase who are beginning to adopt SPF.
- Softfail : the message does not meet a domain's strict

© 2020 LogSat Software

definition of legitimacy, but the domain cannot confidently state that the message is a forgery.

- Fail : the message does not meet a domain's definition of legitimacy.

If the result is "Pass" the email will pass the SPF filter. Behavior for all the other failing results can be customized by the administrators in the SpamFilter GUI by adjusting the settings in the Settings - SPF Filter tab.

In SpamFilter Enterprise, each setting can be configured independently for each email domain.

© 2020 LogSat Software

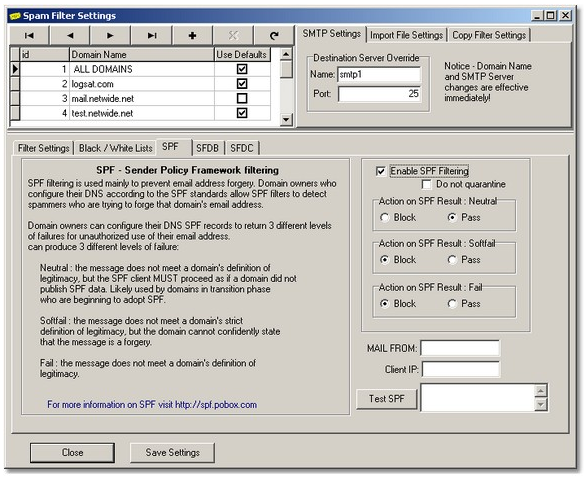


Figure 5

© 2020 LogSat Software

5.7 Bayesian Statistical Filtering

SpamFilter ISP features statistical DNA fingerprinting of incoming emails. The statistical analysis is performed using Bayesian rules. Tokens within incoming emails are scanned and categorized in a corpus file. The content of all new incoming email is fingerprinted and checked against the historical data. If there is a high statistical probability that the email is spam, it is rejected.

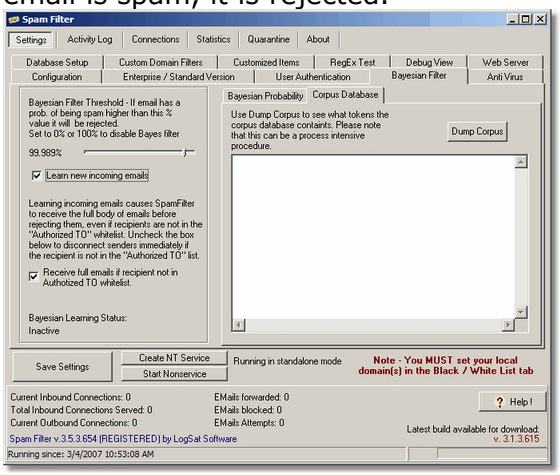


Figure 6

The statistical engine kicks in after 5,000 non-spam and 5,000 spam emails have been received (values customizable by editing the SpamFilter.ini file). This is done to build a valid statistical base to use before emails are rejected. During this period of time, it is critical to avoid false positives. If a good email is quarantined, forcing it's redelivery either thru the web interface or the SpamFilter GUI will "teach" SpamFilter that the fingerprint in that email is a "good" one, and the statistical DNA database will adapt itself to it. It is very important initially to check the quarantine often to force delivery of legitimate email that has been blocked by the "regular" filtering rules.
A slider is used to control the accuracy of the statistical filter. Incoming emails are assigned a probability of being Spam, ranging from 0% (most likely a valid email) to 100% (most likely Spam). Any emails that have a probability of being spam above the value you set will be rejected. Typical threshold values are in the 99.9% range.

- **Bayesian Filter Threshold -** Use this slider to control the accuracy of the statistical filter. Incoming emails are assigned a probability of being Spam, ranging from 0% (most likely a valid email) to 100% (most likely Spam). Any emails that have a probability of being spam above the value you set will be rejected. Typical threshold values are in the 99.9% range.
- **Learn new incoming emails –** If checked, this options allows SpamFilter to continuously monitor incoming emails, thus “learning” about what is spam and what is legitimate email, and to update the statistical database in real-time.
- **Receive full emails if recipient is not in Authorized TO whitelist -** Learning incoming emails causes SpamFilter to receive the full body of emails before rejecting them, even if recipients are not in the "Authorized TO" whitelist. Uncheck the box below to disconnect senders immediately if the recipient is not in the "Authorized TO"

© 2020 LogSat Software

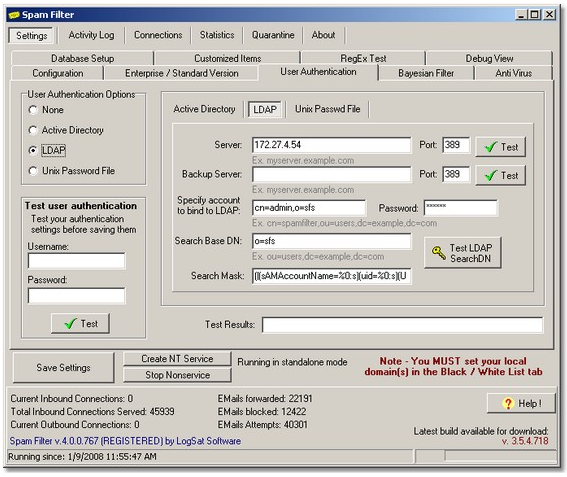
list.

© 2020 LogSat Software

5.8 SMTP Authentication

Many mail servers lack support for SSL and SMTP Authentication. SpamFilter ISP supports both SSL and SMTP AUTH via Active Directory, LDAP, and Unix-style password files. If a user is authenticated, they will be able to bypass all filtering rules and use SpamFilter ISP as a relay to send their outgoing emails.

Administrators can then add support for SMTP Authentication (and SSL) if they have older mail servers that do not have these features.



5.9 Filters per-Domain

In SpamFilter ISP, most filters can be enabled and disabled individually for each domain. In SpamFilter Enterprise, this functionality is greatly enhanced by allowing administrators to specify most settings and most blacklist/whitelist setting individually for each domain.

5.10 Customized Items

Most rejection notices to the remote servers can be customized. In the error string you can embed the following connection-specific parameters:

- %IP% - The IP address of the remote server connecting to SpamFilter
- %Domain% - The MAIL FROM domain name of the incoming email attempt
- %EMailTo% - The recipient of the incoming email attempt
- %EMailFrom% - The sender's email address

If you want to reset a field to its default value you can delete its reference in the SpamFilter.ini file and restart SpamFilter.

© 2020 LogSat Software

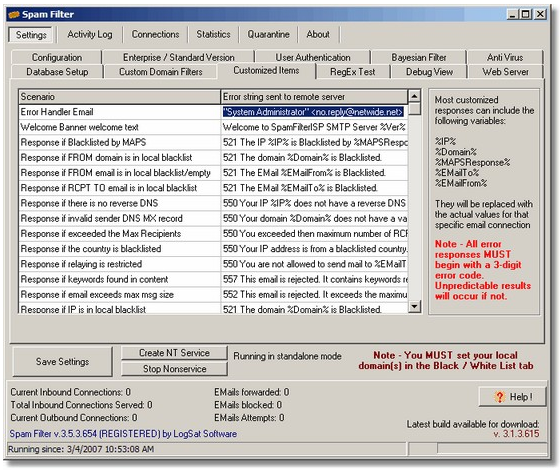


Figure 7

© 2020 LogSat Software

5.11 Filter Order

Filters are processed in a specific order. Once a filter determines an email is spam, all the following filters will be skipped.

Following is the order of the filters. In red are the blacklists, in green the whitelists:

Cached IP blacklist
Greylist
    Whitelisted IP
    Whitelisted Email Address To
    Whitelisted EMail Address From
    Whitelisted Email From Domain
    Whitelisted Auto White List Force Delivery
Allowed Domains
Local IP Blacklist
Local Domain Blacklist
Local Emails Blacklist
Local Emails TO Blacklist
Not in Authorized TO Emails
Country Blacklist
Reject No Reverse DNS
Reject Empty Mail From
Reject Same To From Email address
Reject if Recipient's email in Honeypot email list
Reject if IP in Honeypot-generated auto-ban list
Reject Same To From Domain
Recipient Count > Max RCPTTO
MX Record check
SFDB Filter
SPF Filter
MAPS check
    Keyword Whitelist
SFCD Filter
Blank emails with attachments only
Spam Images in PDFs
Attachment Filter
Keywords
Image Filtering
Bayesian Filtering
SURBL check
Antivirus Plugin

6 Quarantine Database

SpamFilter ISP

SpamFilter ISP does not require a database to filter emails. A database is however

© 2020 LogSat Software

required if you wish to quarantine spam emails that were blocked. The quarantine database allows administrators and end-users to retrieve legitimate emails that were incorrectly detected as spam. When a blocked email is force-delivered, SpamFilter will automatically match the sender with the recipient, so that any future emails from that sender to that recipient will be automatically whitelisted.
More detailed instruction on how to configure a database can be found in the section SpamFilter ISP of this chapter.

SpamFilter Enterprise

SpamFilter Enterprise stores all its configuration settings in a database, making the use of a database mandatory. To ensure maximum uptime in cases where the database server is unavailable, SpamFilter Enterprise caches a copy of all its database settings to local text files. Any changes to the settings stored in the database cause SpamFilter to automatically export the changed settings to its relative cache text file within 5 seconds. The database can be updated even if SpamFilter is not running. In this case, when SpamFilter starts up, it will immediately see the updated tables in the database, and will immediately update its cache files automatically.
More detailed instruction on how to configure a database for SpamFilter Enterprise can be found in the section SpamFilter Enterprise of this chapter.

6.1 SpamFilter ISP

SpamFilter ISP does not require a database to filter emails. A database is however required if you wish to quarantine spam emails that were blocked. The quarantine database allows administrators and end-users to retrieve legitimate emails that were incorrectly detected as spam. When a blocked email is force-delivered, SpamFilter will automatically match the sender with the recipient, so that any future emails from that sender to that recipient will be automatically whitelisted.

Supported Databases

The following database platforms are supported by SpamFilter ISP:

- Microsoft SQL Server 7 or higher
- MySQL v4.1 or higher (v5.1 or higher is required for SpamFilter Enterprise) with MySQL ODBC Connector v3.51.12. **Warning - MySQL ODBC Driver v3.51.14 has a known bug that renders it incompatible with SpamFilter. Please install v3.51.12 or earlier of the driver.**
- Microsoft Access (not supported in SpamFilter Enterprise)
- Oracle 8.x and higher (not supported in SpamFilter Enterprise)

Microsoft Access

An empty MS Access database is included with SpamFilter, and can be found in the \SpamFilter\database directory. As MS Access was not designed for multi-user applications, we do not recommend using Access in a live, high traffic installation. Please note that Access has a 1GB/2GB limit on table sizes, depending on the version of Access being used. When using MS Access, MDAC 2.8 or higher is required on the server running SpamFilter.

MS SQL, MySQL, Oracle

SpamFilter ISP has a simple 1-2-3 step wizard that can create the necessary tables in

© 2020 LogSat Software

Microsoft SQL Server, MySQL and Oracle. An empty database must be created by the DBAs (Database Administrators) before executing SpamFilter's database wizard. If the DBAs wish to manually control the table creation process, the SQL scripts used to create the tables for all 3 platforms are included in the database platform.

MS SQL and MySQL Detailed Configuration Samples

Detailed step-by-step procedures to configure Microsoft SQL Server and MySQL databases can be found in the sections Microsoft SQL Server Database Setup and MySQL Database Setup.

SpamFilter ISP Database Options

The "Database Setup" tab in SpamFilter is used to configure the database options in SpamFilter. The database connection, as with all other options in SpamFilter, can be added/changed while the program runs. The connection string can be defined either by using an Universal Data Link File (.UDL) or by specifying a complete connection string within SpamFilter. We recommend using UDL files since they can be used by the webserver as well without exposing the database password in the web files. Sample UDL files for the 4 supported database platforms are found in the \SpamFilter\database directory. Please ensure that the database user defined in the UDL/connection string must have the proper database rights to be able to create tables.
The following options are available:

- **Days to store quarantined rejected emails -** Normally SpamFilter will reject an email if it considered as spam. You can optionally choose to receive and archive those emails rather than having them lost. The remote server will still receive an error stating that the email was rejected, but you will keep a copy in the quarantine directory for this amount of days. This will allow you to force delivery of legitimate email which could have been filtered. If you enter a **0** in this field quarantine is disabled and email is rejected immediately.
- **Database cleanup interval** - Specifies how often SpamFilter performs a cleanup of the database, removing emails that have passed their retention period.

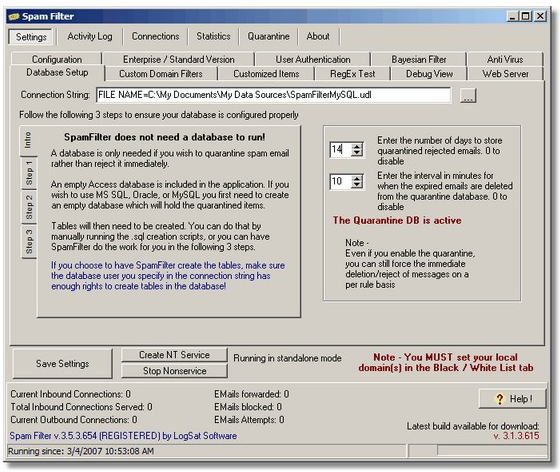


Figure 8

Database Table Details

The three tables used for the quarantine and the privileges needed for them are:

- tblMsgs (contains the message body)

© 2020 LogSat Software

- DB user needs SELECT, INSERT, UPDATE, DELETE rights

o tblQuarantine (contains all details about the email except the content)
- DB user needs SELECT, INSERT, UPDATE, DELETE rights

o tblRejectCodes (contains the current 13 reject categories)
- DB user needs SELECT rights

o tblServers (used when multiple SpamFilters share the same database)
- DB user needs SELECT, INSERT, UPDATE rights

o tblLogins (used to store usernames and passwords for users who access the web-based quarantine)
- DB user needs SELECT rights (the account used by the APS/PHP pages will need INSERT, UPDATE rights as well)

© 2020 LogSat Software

6.2 SpamFilter Enterprise

SpamFilter Enterprise requires the use of a database. To maximize SpamFilter's availability, SpamFilter maintains a local file-based copy of its database settings locally on the server. This allows SpamFilter Enterprise to function even if the database is offline. To achieve this goal, SpamFilter Enterprise relies on triggers available only in certain database servers. SpamFilter Enterprise does not support Microsoft Access, Oracle, and older versions of MySQL and Microsoft SQL Server.

Supported Databases

The following database platforms are supported by SpamFilter Enterprise:

- Microsoft SQL Server 2000 or higher
- MySQL v5.1 or higher with MySQL ODBC Connector v3.51.12. **Warning - MySQL ODBC Driver v3.51.14 has a known bug that renders it incompatible with SpamFilter. Please install v3.51.12 or earlier of the driver.**

Prerequisites

SpamFilter Enterprise will run in "Standard" mode when the database is not configured (the default mode). Before being able to switch functionality to the "Enterprise" mode, SpamFilter requires that a supported database platform be configured using the Database Setup tab.

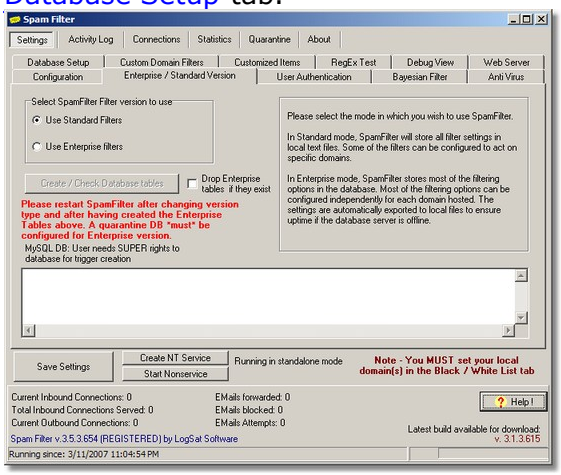


Figure 9

Enterprise Database Configuration

Once the database has been configured to use the SpamFilter "Standard" tables, the "Enterprise Mode" can be enabled. To proceed, perform the following 4 steps on the "Enterprise / Standard Version" tab, as shown on the screenshot in Figure 10.

1. Click on "Use Enterprise Filters"
2. Click on the "Save Settings" button.
3. Click on "Create / Check Database tables" button to have SpamFilter create the necessary Enterprise tables. The database script commands being executed, and any errors that may result, will be displayed in the memo field underneath the button. We recommend scrolling thru the results to see if any major errors

© 2020 LogSat Software

appear in the results.

4. Once the script in the above step has completed, restart SpamFilter.

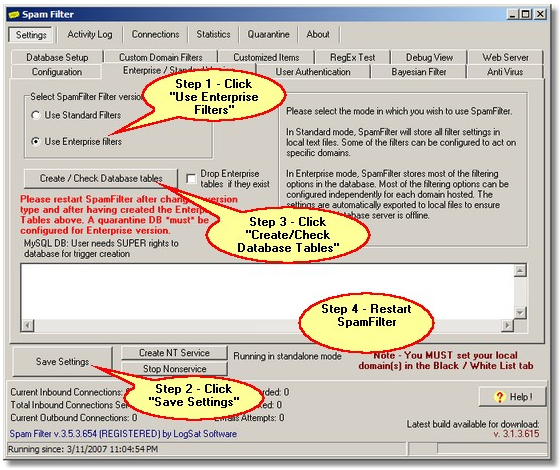


Figure 10

© 2020 LogSat Software

6.3 Microsoft SQL Server Database Setup

The steps to configure SpamFilter ISP to use a Microsoft SQL Server database are:

1. Install Microsoft SQL Server on a server (it can be installed on the same server running SpamFilter).
2. Create an empty database in SQL Server. The samples below assume this database to be named "spamfilter", but you can use any other naming convention
3. In SQL Server, create a new SQL user and assign it dbo privileges for the "spamfilter" database. After the necessary tables have been created, the privileges can be reduced to allow this user only read/write data.
4. Configure the database connection in SpamFilter to point to the above database. The simplest way to proceed is to use a UDL file (Microsoft's Universal DataLink) to define the connection to the SQL server. In the SpamFilter\database directory you'll find several .UDL files for each database platform. Please double-click on the one for SQL Server, SpamFilterMSSQL.udl. The correct OLE DB Provider should already be selected in the "Provider" tab. Under the "Connection" tab enter the name of your SQL server, and provide the username/password just created. Ensure you check the "Allow saving password" box so the credentials are saved in the UDL file. Please see the screenshot in Figure 13 for sample settings. Click on "Test Connection" to ensure the UDL file works.

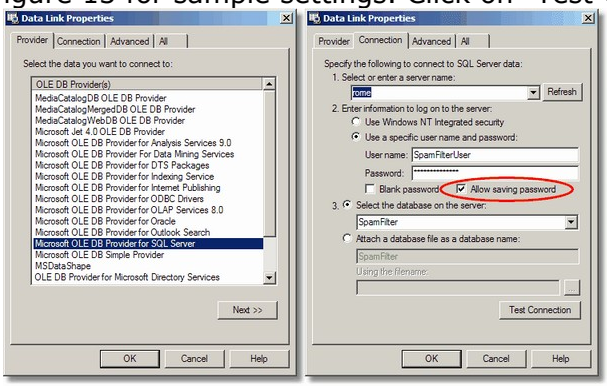


Figure 13

5. In SpamFilter, under the "Settings - Quarantine DB" tab, click on the "..." ellipse button to select a Connection String. In the box that follows, select "Use Data Link File" and select the SpamFilterMSSQL.udl you customized before. See screenshot in Figure 14 for sample settings.
6. Use the 1-2-3 step wizard in SpamFilter's Quarantine DB tab to create the necessary tables in SpamFilter. On step 3, ensure to click on the "Save & Activate Conn String" button.
7. Click on the "Save Settings" button.

© 2020 LogSat Software

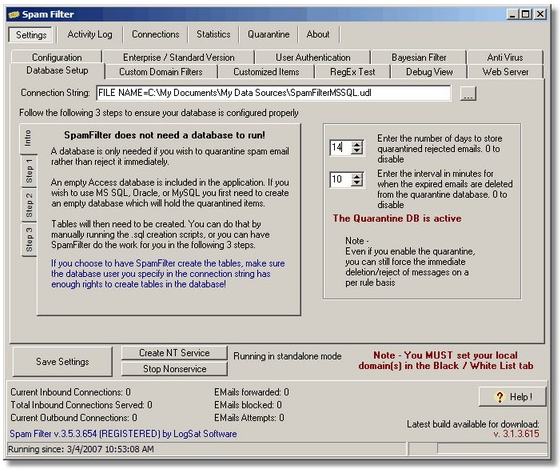


Figure 14

© 2020 LogSat Software

6.4 MySQL Database Setup

The steps to configure SpamFilter ISP to use a MySQL database are:

1. Install MySQL on a server (it can be installed on the same server running SpamFilter). On Windows platforms we recommend installing the "MySQL Administrator" tool as it allows easier administration for MySQL.
2. Install the MySQL ODBC Driver v3.51 (do not use any beta drivers). **Warning - MySQL ODBC Connector Driver v3.51.14 has a known bug that renders it incompatible with SpamFilter. Please install v3.51.15 or later, or v3.51.12 or earlier of the driver.**
3. Create an empty database in MySQL. The sample screenshots assume this database to be named "spamfilter", but you can use any other naming convention.
4. In MySQL, create a new user ("SpamFilterUser" in the sample screenshots), and assign it all privileges for the "spamfilter" database. Please see the screenshot in Figure 9 for a sample.

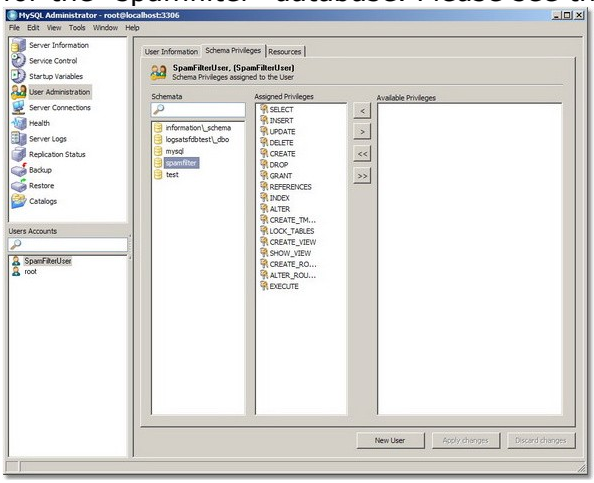


Figure 9

5. In the ODBC Data Sources on the server, create a DSN for the above database using the MySQL ODBC Driver. See the screenshot in Figure 10 for a sample.
6. Configure the database connection in SpamFilter to point to the DSN above. The simplest way to proceed is to use a UDL file (Microsoft's Universal DataLink) to define the connection to the MySQL server. In the SpamFilter\database directory you'll find several .UDL files for each database platform. Please double-click on the one for MySQL, SpamFilterMySQL.udl. The correct OLE DB Provider should already be selected in the "Provider" tab. Under the "Connection" tab select the DNS created in step 4 above. Please see the screenshot in Figure 11 for sample settings. Click on "Test Connection" to ensure the UDL file works. **Please ensure that the "Return Matching Rows" option in the Advanced settings has been checked**.

© 2020 LogSat Software

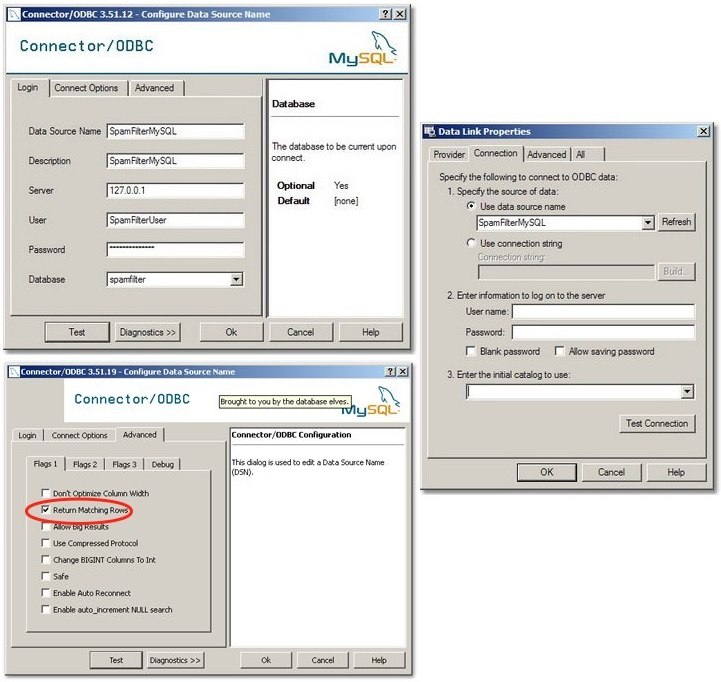


Figure 10 / Figure 11

7. In SpamFilter, under the "Settings - Quarantine DB" tab, click on the "..." ellipse button to select a Connection String. In the box that follows, select "Use Data Link File" and select the SpamFilterMySQL.udl you customized before. See screenshot in Figure 12 for sample settings.
8. Use the 1-2-3 step wizard in SpamFilter's Quarantine DB tab to create the necessary tables in SpamFilter. On step 3, ensure to click on the “Save & Activate Conn String” button.
9. Click on the “Save Settings” button.

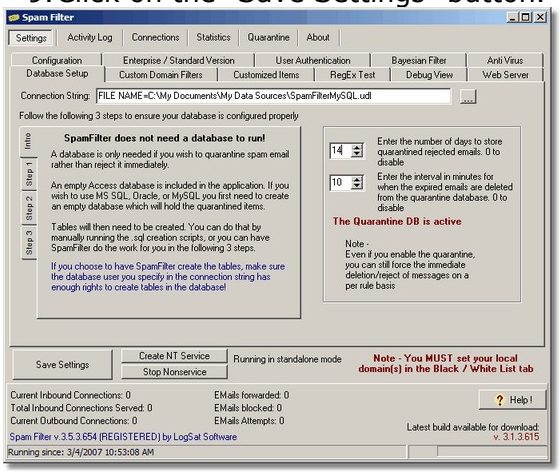


Figure 12

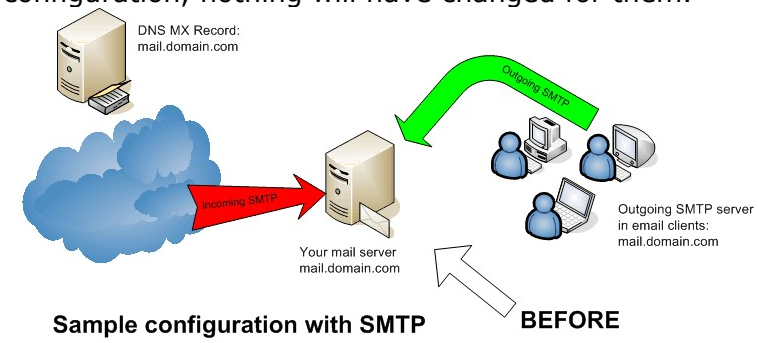
© 2020 LogSat Software

7 Network Configurations Samples

To detect spam accurately, the network must be configured so that incoming internet email is routed to SpamFilter. SpamFilter should be the first application/server to receive and process incoming internet emails in your network.
If SpamFilter is placed behind a firewall, ensure that the firewall is configured so that SpamFilter sees the internet IP address, and not a translated address, for the remote servers.

7.1 SMTP servers directly connected to Internet

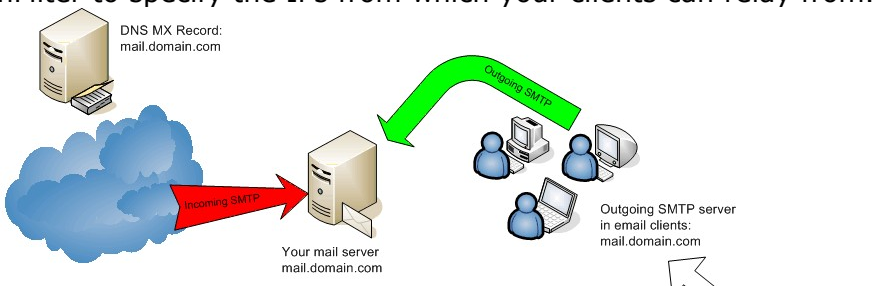
To implement SpamFilter with minimal impact you would do the following.
Configure SpamFilter to listen on a different IP, for example on mail2.domain.com. This can be done by installing SpamFilter either on a separate server or the same server provided it has multiple IP addresses assigned to it. Reconfigure the MX record to point to mail2.domain.com. SpamFilter will now be receiving all internet email and will then forward legitimate emails to your SMTP server at mail.domain.com. Your existing customers still have mail.domain.com for SMTP server in their email client configuration, nothing will have changed for them.



© 2020 LogSat Software

7.2 SMTP server with single IP address connected to Internet

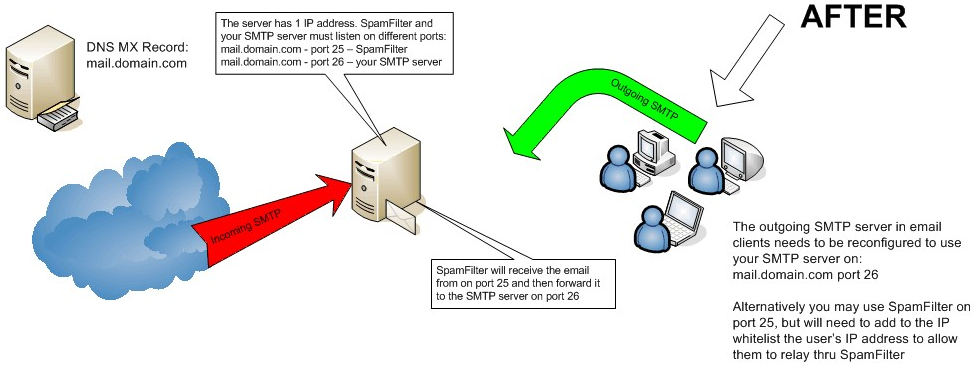
In this configuration SpamFilter is installed on the same server as your SMTP software. The server only has one IP address available, SpamFilter will need to be configured to listen on port 25 so it can accept email traffic. Your SMTP software will need to be reconfigured to listen on a different port, for example port 26. SpamFilter will then forward clean emails to your SMTP server on port 26.All your email clients will need to be reconfigured so that their "Outgoing SMTP server" now points to your SMTP server on port 26. Alternatively, you may choose to let your email clients relay thru SpamFilter. Please note that in order to do so, you will need to configure IP whitelists in SpamFilter to specify the IPs from which your clients can relay from.



Sample configuration with a single SMTP server and a single IP connected to Internet

BEFORE

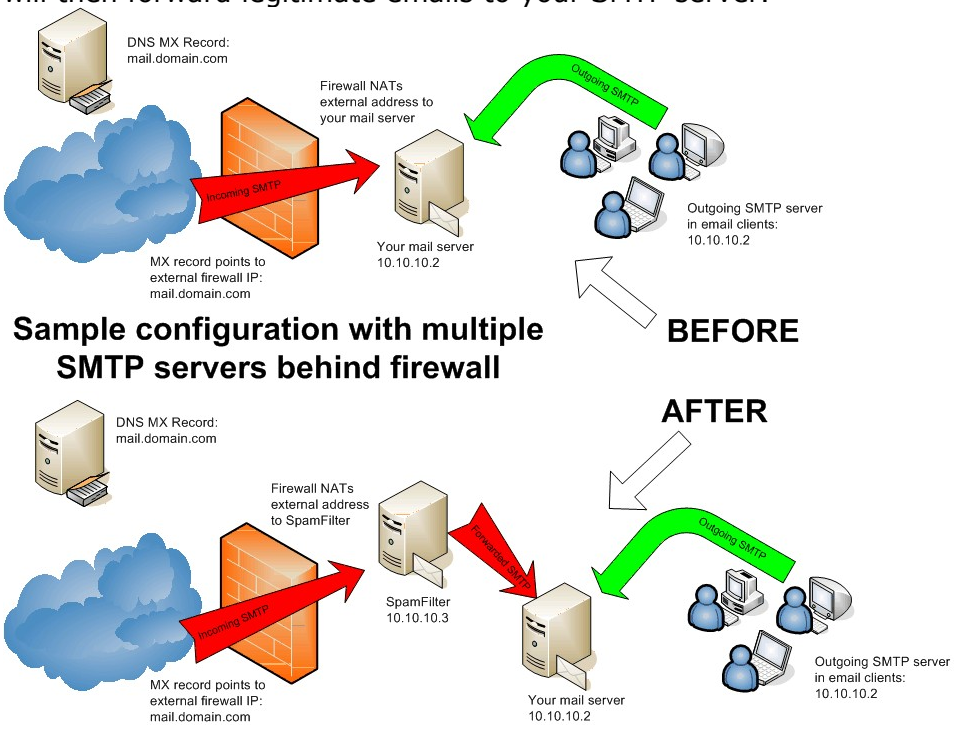
AFTER



© 2020 LogSat Software

7.3 SMTP servers behind Firewall

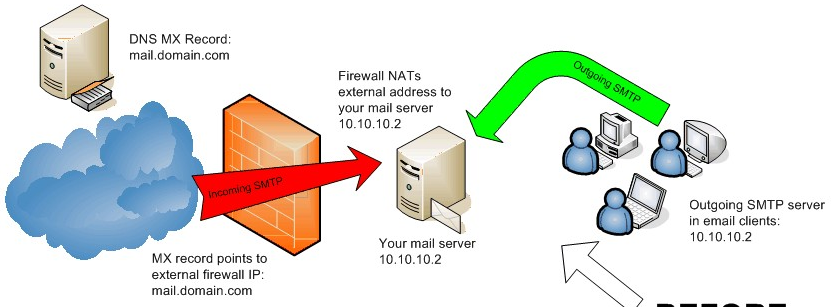
This is one of the simplest configurations to implement. Install SpamFilter on a standalone server. Reconfigure the firewall so that it routes the external IP address to the IP assigned to SpamFilter. SpamFilter will now be receiving all internet email and will then forward legitimate emails to your SMTP server.



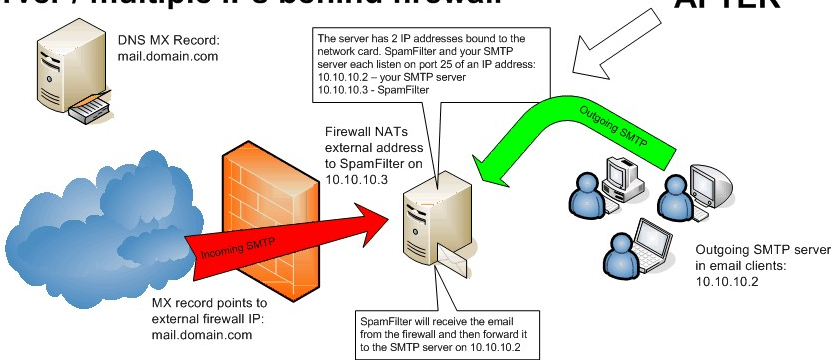
© 2020 LogSat Software

7.4 Single SMTP multihomed server behind Firewall

Configure SpamFilter to listen on a different IP than your SMTP server. This can be done by assigning additional IP address to the server. Reconfigure the firewall so that it routes the external IP address to the IP assigned to SpamFilter. SpamFilter will now be receiving all internet email and will then forward legitimate emails to your SMTP server.



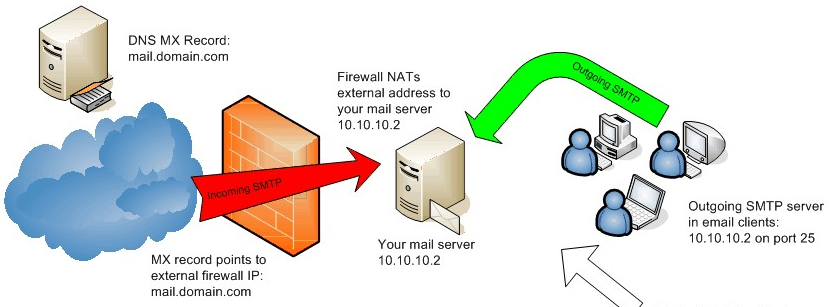
Sample configuration with single SMTP server / multiple IPs behind firewall



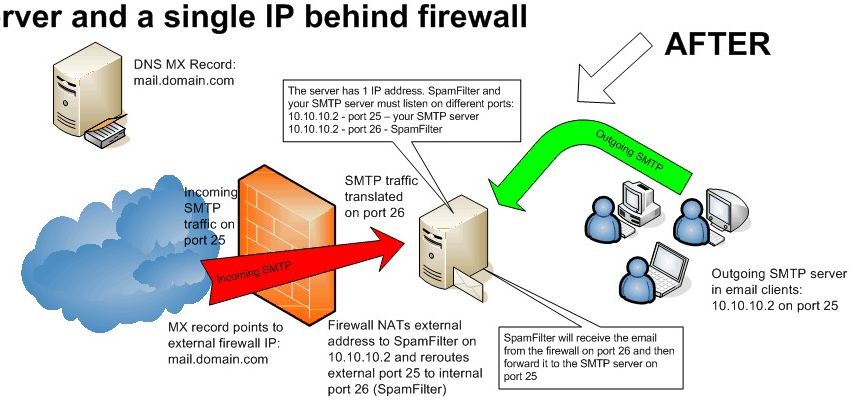
© 2020 LogSat Software

7.5 Single SMTP server behind Firewall

In this configuration SpamFilter is installed on the same server as your SMTP software. The server only has one IP address available, SpamFilter should be configured to listen on a port other than 25 so it does not conflict with your SMTP Server, for example port 26. Your SMTP software will not need be reconfigured. Reconfigure the firewall to perform port mapping so that it routes the external SMTP traffic from port 25 to port 26 on the IP assigned to SpamFilter. If the firewall does not support port translation, this configuration can't be implemented. Please refer to the "Single SMTP server with a single IP address directly connected to Internet" example above for an alternate solution.



Sample configuration with a single SMTP server and a single IP behind firewall



© 2020 LogSat Software

8 Running SpamFilter

8.1 SpamFilter Service / Console Application

SpamFilter can run in two different ways.

- As a Windows service (the most common - SpamFilterSvc.exe).
- As a standalone application using SpamFilter.exe. This is mostly used for troubleshooting and testing purposes.

If SpamFilter is already running as a service, subsequently running the standalone application (SpamFilter.exe) will open a new instance of SpamFilter, *it will not be the GUI for the service*. This will most likely create a conflict, as two applications cannot bind to the same port on a server.

Please note that the SpamFilter service does show display a GUI (launched from an icon in the tray bar), but if you are accessing a Windows 2000 server remotely using Terminal Services will not be able to display the GUI. This is because Terminal Services in Windows 2000 is not able to display the server's physical console.

Looking at the physical server's screen, or using a product like PCAnywhere, DameWare, VNC etc that displays the actual screen will reveal the console.

Microsoft fixed this limitation in Windows 2003. In this version of the operating system, Terminal Services allows RDP clients to connect to the server's console.

Some Terminal Services clients have a checkbox in their settings that forces them to connect to the console. In the Remote Desktop client that ships with Windows XP, Microsoft (in)conveniently decided to not make this checkbox available. In this case, to view the server's physical console, you'll need to invoke Remote Desktop from the command line as follows:

mstsc -console

8.2 Running multiple SpamFilters

SpamFilter is easily implemented in a load-balanced environment. SpamFilter can be installed on multiple servers, all sharing the same quarantine database. Each instance of SpamFilter will "know" which quarantined spam emails it own, and will process them accordingly.

To have multiple SpamFilters connect to the same database, simply configure each SpamFilter to point to the same database. Each instance will add itself, automatically, to the database so it can share it with the others. That's all...

SpamFilter ISP

SpamFilter ISP maintains all its configurations in local text file. The file SpamFilter.ini contains the main settings for SpamFilter. When using multiple SpamFilters, the SpamFilter.ini file will be the same on all servers. The only exception is if you configure one or more SpamFilter to listen for SMTP traffic on a specific IP on the server., rather than the default "bind to all IPs". In this case, the ini files would be different.

© 2020 LogSat Software

The various blacklist / whitelist / keyword entries are stored in individual, user-defined files. We recommend placing these files in a subdirectory for better organization. In a load-balanced environment, these files will usually be the same for all servers. Having them in a separate directory allows them to be more easily replicated across the multiple servers.

SpamFilter continuously monitors all its configuration files, and if any of them is changed by an external application, SpamFilter will automatically reimport all the settings within a few seconds.

SpamFilter Enterprise

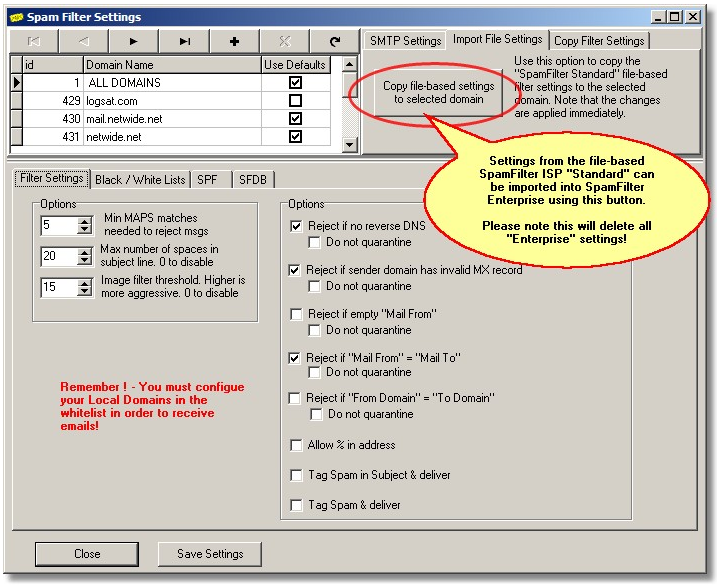
SpamFilter Enterprise maintains all its configurations in a database, making it simpler for multiple SpamFilter servers to share the same data. Simply configure each instance of SpamFilter to use the same database as described in the section Quarantine Database - SpamFilter Enterprise. When SpamFilter Enterprise is started, it will be automatically added to the list of servers in the database, no other configurations are necessary.

© 2020 LogSat Software

9 Upgrading SpamFilter ISP to SpamFilter Enterprise

When SpamFilter's operating mode is switched from "SpamFilter ISP "Standard" to "SpamFilter Enterprise" and SpamFilter Enterprise is restarted, the initial filter settings will assume default values.

Any existing "Standard" settings will need to be re-imported into SpamFilter Enterprise using the "Copy file-based settings" button as shown in the screenshot below.



9.1 Upgrading to SpamFilter Enteprise "offline"

Administrators managing installations with several domains and/or complex filters may want to configure and test the upgrade from SpamFilter ISP to SpamFilter Enterprise before proceeding with the upgrade.

The following section describes a methodology that can be used to configure your SpamFilter Enterprise settings before actually performing the upgrade, keeping your current SpamFilter ISP running at all times, thereby eliminating any downtime.

SpamFilter Enterprise (SFE) was designed to share the same database used by SpamFilter ISP "Standard" (SFI). Furthermore, the "Enterprise" (SFE) tables are completely independent from the "Standard" (SFI) tables. This allows administrators to configure all the "Enterprise" settings without interfering with the "Standard" settings.

This means that, for example, you run a second instance of SpamFilter "standard", and perform the upgrade on this second instance, leaving your live production instance "as-is". Upgrading the second instance will create all the SFE-related tables in the live production database, but the two will not interfere with each other. You can then proceed to configure and test all your SFE settings without impacting your "live" SpamFilter installation.

© 2020 LogSat Software

After your SFE configuration is completed and are satisfied with its testing, the database will contain the correct settings that SFE will use. At this point, when you're ready, you can proceed to change the operating mode of your "live" copy of SFI to SFE, and restart SpamFilter. Your "live" SpamFilter will now immediately work by reading the SFE tables that you've already configured, and your upgrade downtime will be limited to the few seconds it takes to stop and restart SpamFilter's service.

To proceed this way, you will then have to configure a second instance of SpamFilter. Since SpamFilter was developed so that all files reside in the installation directory, its configurations files are stored in text files, and no settings are stored in Windows' registry, this process is rather simple.

We suggest to copy your entire SpamFilter installation directory to a different location on your server. You do not need to stop SpamFilter during the copy process, as the only files that can be locked are possibly some temp files used to spool emails. You may want to exclude the "SpamFilter\logfiles" and the "SpamFilter\queue" directory from the copy process to save some time and avoid file sharing warnings.

In the sample below, we'll assume your "live" SpamFilter installation directory is:
c:\program files\SpamFilter
and that you will be installing your second temporary instance to:
c:\program files\SpamFilterEnterprise

After copying the directory, you will need to edit the only two configuration files and update any directory path they may contain. These two files are:
c:\program files\SpamFilterEnterprise\SpamFilter.ini
c:\program files\SpamFilterEnterprise\Domains\SFI\Filters.ini (this file may not exist if you have not upgrade to SpamFilter v3.5 yet).

In the SpamFilter.ini file, look for the line:
ListenPort=25

This is used to tell SpamFilter on which port to listen for incoming SMTP traffic. You will need to change it so your second instance won't interfere with your live one. Assuming your server is not using port 26, please change the entry above to something like (you can choose a different port if you wish):
ListenPort=26

After changing the port number, while not strictly necessary, as a precaution you may want to update the *SpamFilter.ini* and the *Filters.ini* files above so that they point to the blacklist/whitelist files in your new directory, not to those in your live directory. To do this, simply do a "search and replace" for all text in your SpamFilter.ini and Filters.ini files, looking for the string "*c:\program files\SpamFilter*" and replace it with "*c:\program files\SpamFilterEnterprise*" (note the trailing "\").

Your second instance of SpamFilter is now configured exactly like your first one. You can now double-click on the "c:\program files\SpamFilterEnterprise\SpamFilter.exe" executable to start your second instance of SpamFilter. Please make sure there are no errors being displayed in the "Activity Log" tab upon startup, and if so, you may continue.

To upgrade this second instance to SpamFilter all you need to do is to:

© 2020 LogSat Software

- Go to the "Settings" - "Enterprise / Standard Version" tab.
- Click on "Use Enterprise Filters".
- Click on the "Save Settings" button.
- *Click on the "Create / Check Database tables" button. This will create the SFE tables in the database, and this process should not affect your "live" SFI running in the background.*
- Close the two windows for this second instance of SpamFilter, which will shut down the program.
- Once more, double-click on the "c:\program files\SpamFilterEnterprise\SpamFilter.exe" executable to restart SpamFilter Enterprise.

You can now safely make any configuration changes to your second instance running SpamFilter Enterprise. All filter settings will be saved to the common database, without affecting your live installation. When you are ready to upgrade your live copy of SpamFilter, you will only have to execute the following steps (note that they are similar to the above steps, but are missing the italicized one above):

- Go to the "Settings" - "Enterprise / Standard Version" tab.
- Click on "Use Enterprise Filters".
- Click on the "Save Settings" button.
- Stop and restart SpamFilter's service.

10 Antivirus Plugin

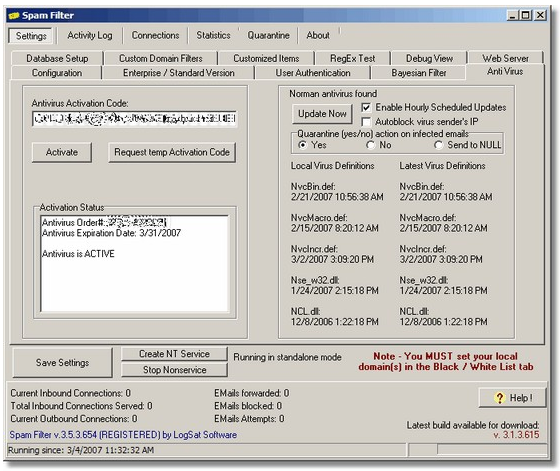


Figure 15

LogSat Software has partnered with NORMAN (www.norman.com) to provide optional antivirus protection for email traffic.
The antivirus plug-in is be available for purchase separately from SpamFilter ISP and is an optional component. Unlike SpamFilter ISP's licenses, the antivirus plug-in is offered as a subscription service with a yearly subscription fee of $400.
An Activation Code is required to enable the antivirus plug-in. A 15-day trial Activation Code can be obtained from SpamFilter's GUI on the Settings - Antivirus tab. The code is required only for the antivirus plug-in activation. SpamFilter (both retail and free versions), will continue to work and stop spam even without the antivirus plug-in.

© 2020 LogSat Software

Please note that only one request for a trial code will be honored per installation.
Technical notes - SpamFilter can run with or without the antivirus plug-in. When SpamFilter starts, it will check for the plug-in files. If they are found, antivirus support will automatically be enabled. We recommend installing the antivirus plug-in after installing SpamFilter. Restart SpamFilter after installing the plug-in to activate it.
The antivirus plug-in is a set of 3 DLLs that are to be installed in the SpamFilter directory: dwnse.dll, ncl.dll and nselapi.dll. In addition to those files, the Norman scan engine needs to be present. If a Norman product is not already installed on the server, the installer adds the necessary files (including the virus definitions) in the SpamFilter\nse directory.
There are 2 additional DLL's that are placed in the SpamFilter directory by the install program: libeay32.dll and ssleay32.dll. These DLLs are used by the antivirus plug-in. If performing a manual install please make sure you copy these files to the SpamFilter directory.

© 2020 LogSat Software

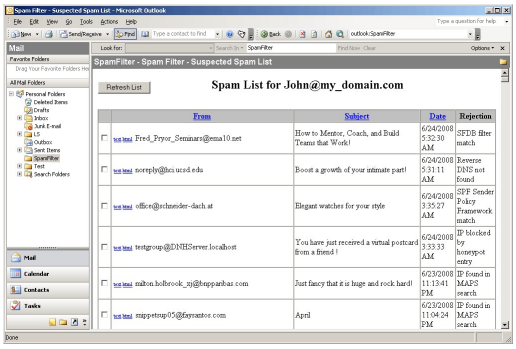
11 Log Analysis & Statistics

SpamFilter ISP log files can be parsed by Sawmill (www.sawmill.net) an excellent log analysis tool. Sawmill generates reports of email traffic by IP, domain, country, sender and recipient, action taken on messages and much more. In the SpamFilter\Database directory you will find the Sawmill plug-in file **SpamFilterISP**. If your copy of Sawmill 6.5 or higher does not recognize SpamFilter ISP's log format, simply copy that file in the Sawmill\LogAnalysisInfo\LogFormats directory to allow it to read SpamFilter ISP logs.

12 Web Interface for End-User Quarantine Access

End-users can access their quarantined emails via a web interface. This allows them the ability to view the spam emails that were blocked by SpamFilter, and to force the delivery of any valid email that was blocked by mistake. It is to be noted that once a user forces the delivery of an email from the quarantine, SpamFilter will automatically match the sender with the recipient. From then on, all future emails from that sender to that recipient will be automatically whitelisted. This will greatly reduce the risk of blocking valid emails in the future.

For corporations using Microsoft Outlook email clients, a very nice feature is the ability to display a "SpamFilter" folder within the Outlook client. This web-enabled folder then allows the end users to see their quarantined emails directly within their Outlook client without the need of an external web browser. From Outlook they can then view/deliver any emails in the quarantine.



12.1 Web Server Configuration

During SpamFilter's installation, the installer provides an option to install an ASP-based sample web interface to access the quarantine area. This requires Microsoft IIS to be installed on a server, and the Active Server Pages extensions to be enabled.

The same sample ASP pages, along with a PHP version of the same application, is also available for download on our website.

The webserver does not need to communicate with the server where SpamFilter is installed on. The web server instead does need to connect to the database server. We recommend using an UDL file for the database connection in the ASP/PHP code, as you are able to place this file in a secure location on the webserver, outside the public web area, making it harder for intruders to gain access to it. The UDL file can contain the database password, so you will not have to store it in the web pages.

Configuration

The only item that needs to be configured in the web interface is the database connection. To proceed, simply edit the db_Connect.asp web page, and modify the line:

UDLPath = "c:*EditThisPath*\SpamFilter.udl"

to reflect the correct path to the UDL file on the webserver that is used to connect to the database. Sample UDL files for the various database platforms are located in the \SpamFilter\database installation directory.

User Authentication for Web Access to Quarantine

The sample ASP-based web interface provided allows users to self-register for the quarantine access. Users can enter their email address and a random password will be generated and emailed to them. They will then be able to login with their email address access their quarantine area.

Customizing User Authentication

An important part in using the web interface is choosing a way to authenticate users. We provide a **tblLogins** table in the database that can store a list of Email addresses and passwords. Our sample authenticate.asp and authenticate.php pages perform authentication based on that table. You can choose your own authentication schema and create your own pages to authenticate in other ways. At the end, ensure that the authenticated email address will be stored in a session variable. The ListSpam and ResolveSpam pages list and deliver the emails belonging to the address stored in the session variable.

When an end user forces the delivery of a quarantined email to his mailbox, the sender of that email will be whitelisted so that the number of false-positives (good emails wrongly classified as spam) is reduced. The list of user-created entries is stored in the file AutoWhiteListForceDelivery.txt. The whitelisting is on a per-user basis, meaning that a sender is whitelisted only when he sends emails to that specific recipient. This will prevent a user mistakenly whitelisting a spammer, who could then send spam to all of your users.

© 2020 LogSat Software

The tblQuarantine has a *Deliver* field and an *Expire* field with default values of 0. Changing the *Deliver* field to 1 will cause SpamFilter to deliver that email within 10 seconds. Changing the *Expire* field to 1 will cause SpamFilter to erase that email from the database within 1 hour. The web pages simply update these two fields to deliver and delete the emails.

Limitations in the Evaluation version of SpamFilter

Note that in the free version of SpamFilter, the web interface will not deliver emails to the recipients!

12.2 Microsoft Outlook Configuration

Configuring Microsoft Outlook to display SpamFilter's quarantined emails is very simple.

To begin, the user needs to register for access via the quarantine website to receive their password. Once the password is received, all that is required is to create a new folder within the Outlook client, and then point this folder to the URL used to access the quarantine area. In the URL the user should manually enter their email address and their password as can be seen from the screenshots below. This only needs to be done once. From the on, they will be able to view their spam by simply clicking on their "SpamFilter" folder.

Image 1

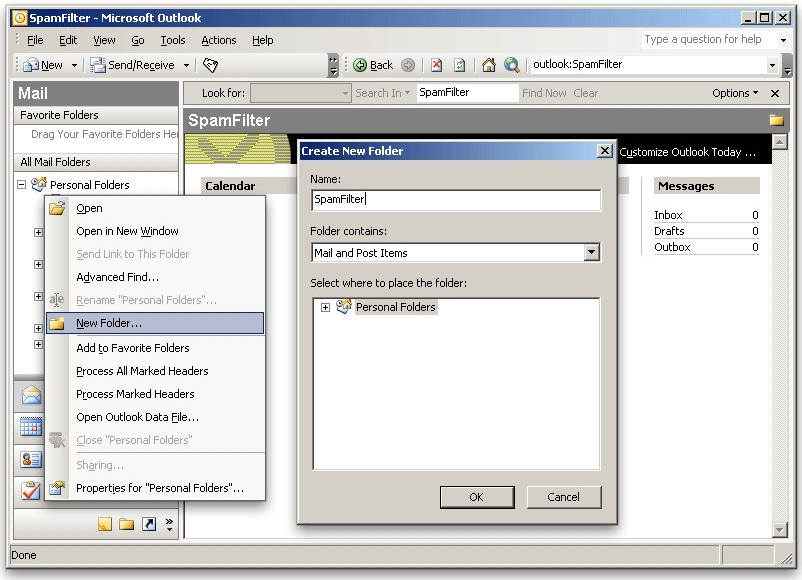


Image 2

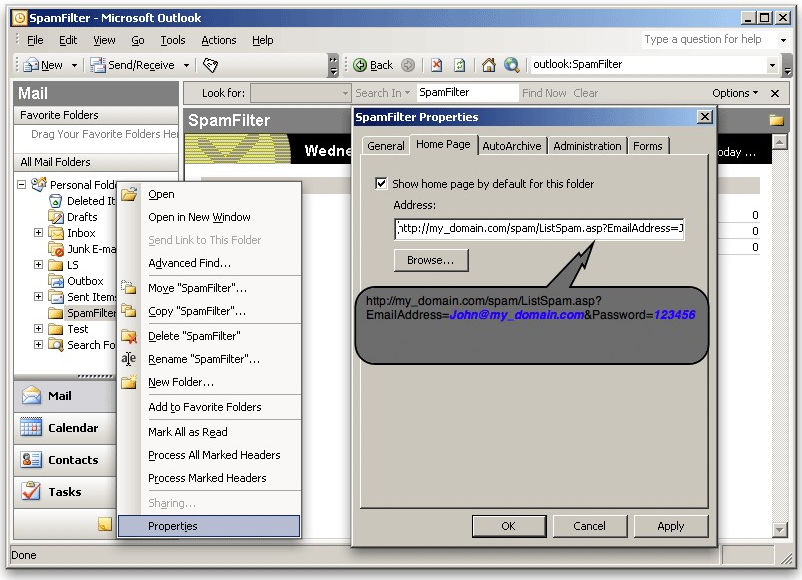
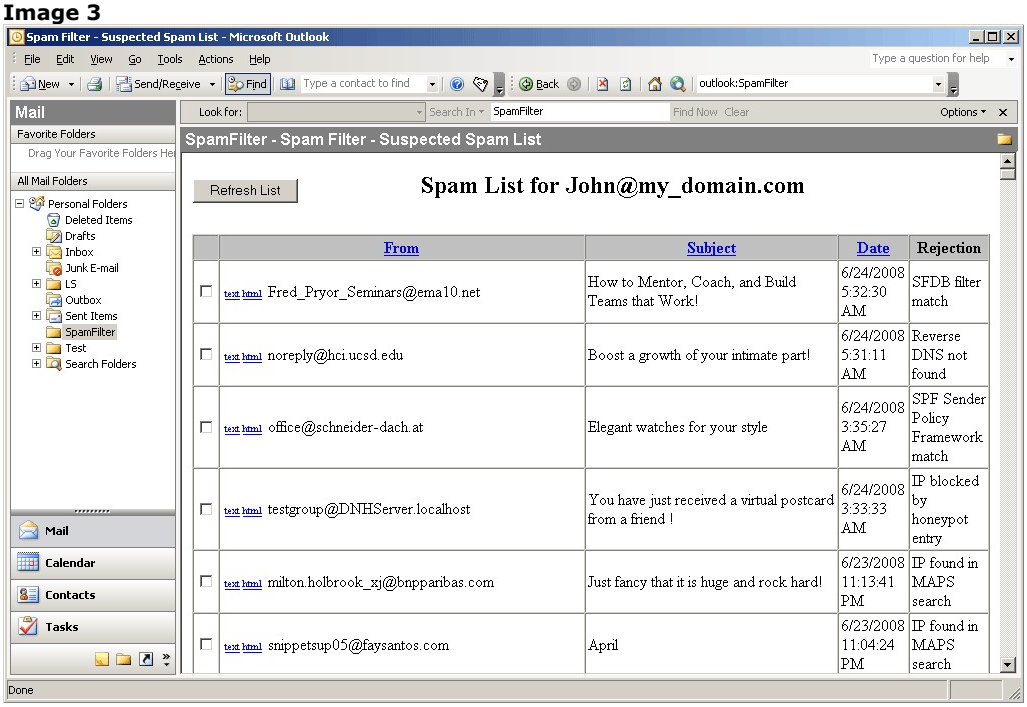


Image 3



13 SSL Certificates

SpamFilter requires the following certificates to be placed in SpamFilter's root directory when accepting SSL connections via SMTP:

- root.pem: Certificate Authority (CA) certificate
- cert.pem: x.509 certificate, signed by the CA
- key.pem: x.509 private key

Included with SpamFilter's distribution are sample certificates issued by LogSat Software. While they are fully functional and allow encrypted communications, they are

© 2020 LogSat Software

signed by our internal Certificate Authority (CA). As this CA is not trusted by browsers and mail client software, using them will often cause security warnings in client software.

Administrators can purchase SSL certificates from commercial entities like Verisign or Thawte, which will eliminate any security warnings. Administrators can also select to issue their own certificates, using a CA they trust. In these cases, the certificates will have to be converted to .pem format if they are issued in other formats. The OpenSSL utility from www.openssl.org can be used for both of these purposes. Pre-compiled binaries for OpenSSL for Windows can usually be found at www.openssl.org/related/binaries.html.

13.1 Creating and self-signing your own certificate

To create and self-sign a certificate using OpenSSL, you can issue, from a MSDOS command prompt, the following command:

openssl req -new -x509 -keyout key.pem -out cert.pem -nodes -days 3650

When following the prompts to generate the certificate, please note that when asked for the "Common Name", you should enter the DNS name of the server you are installing the certificate on, ex: mail.logsat.com.

Now that you created and signed your own certificate, you will have the files key.pem and cert.pem. SpamFilter also needs the CA certificate. As you signed your own certificate, the CA is certificate will be the same as your public certificate. So simply copy/paste the file cert.pem to root.pem. Place all three files (key.pem, cert.pem and root.pem) in the SpamFilter directory and restart SpamFilter to activate the certificates.

13.2 Exporting an existing commercial SSL certificate

If you already purchased a commercial SSL certificate, you need to export it into the .pem format used by SpamFilter. This procedure assumes that you have already received your key and certificate pair from some Certificate Authority (like Verisign or Thawte) and that you have installed them in Microsoft Internet Explorer in the Personal Certificates Store.

- Export Certificate
  - Select the certificate and export it as a .pfx file (Personal Exchange Format). You may optionally protect it with a password.
- Convert .pfx to .pem using OpenSSL:
  - Issue the following command from a MSDOS prompt:

    openssl.exe pkcs12 –in <your file>.pfx –out <your file>.pem

    Openssl.exe will prompt you for a password. Enter it if you used one, or leave it blank if you did not specify one. It will also prompt you for a new password for the .pem file. This is optional, but if you protect it with a password you will need to enter the password in the SpamFilter.ini file (SSLCertificatePassword setting). Use the "-nodes" option in the above command to avoid specifying a new password.
- Split the .pem file into the private and public key files

© 2020 LogSat Software

- If you examine the new .pem file just created with a text editor, you will notice that it consists of two parts. The two parts are the private key and the certificate (public key). There is also some addition information included. SpamFilter requires that this information be separated into separate files.
- Key.pem
  - Create a blank text file with Notepad, name it key.pem, and paste everything between and including these two statements:

    -----BEGIN RSA PRIVATE KEY-----

    -----END RSA PRIVATE KEY-----
- Cert.pem
  - Create a blank text file with Notepad, name it cert.pem, and paste everything between and including these two statements:

    -----BEGIN CERTIFICATE-----

    -----END CERTIFICATE-----
- Root.pem
  - The final file that SpamFilter requires is the Certificate Authority certificate file. You can obtain this from the Internet Explorer in Trusted Root Certificate Authority dialog. Select the Authority that issued your certificate and export it in Base64 (cer) format. This format is also the same as PEM format so after export simply rename the file to root.pem.

© 2020 LogSat Software

14 Regular Expressions (RegEx)

Syntax of Regular Expressions

Introduction

Regular Expressions are a widely-used method of specifying patterns of text to search for. Special **metacharacters** allow You to specify, for instance, that a particular string You are looking for occurs at the beginning or end of a line, or contains **n** recurrences of a certain character.

Regular expressions look ugly for novices, but really they are very simple (well, usually simple ;)), handy and powerful tool.

Simple matches

Any single character matches itself, unless it is a **metacharacter** with a special meaning described below.

A series of characters matches that series of characters in the target string, so the pattern "bluh" would match "bluh'' in the target string. Quite simple, eh ?

You can cause characters that normally function as **metacharacters** or **escape sequences** to be interpreted literally by 'escaping' them by preceding them with a backslash "\", for instance: metacharacter "^" match beginning of string, but "\^" match character "^", "\\" match "\" and so on.

Examples:

foobar	*matches string 'foobar'*
\^FooBarPtr	*matches '^FooBarPtr'*

Escape sequences

Characters may be specified using a **escape sequences** syntax much like that used in C and Perl: "\n'' matches a newline, "\t'' a tab, etc. More generally, \xnn, where nn is a string of hexadecimal digits, matches the character whose ASCII value is nn. If You need wide (Unicode) character code, You can use '\x{nnnn}', where 'nnnn' - one or more hexadecimal digits.

\xnn	*char with hex code nn*
\x{nnnn}	*char with hex code nnnn (one byte for plain text and two bytes for Unicode)*
\t	*tab (HT/TAB), same as \x09*
\n	*newline (NL), same as \x0a*
\r	*car.return (CR), same as \x0d*
\f	*form feed (FF), same as \x0c*
\a	*alarm (bell) (BEL), same as \x07*
\e	*escape (ESC), same as \x1b*

Examples:

foo\x20bar	*matches 'foo bar' (note space in the middle)*

© 2020 LogSat Software

\tfoobar *matches 'foobar' predefined by tab*

Character classes

You can specify a **character class**, by enclosing a list of characters in [], which will match any **one** character from the list.

If the first character after the "[" is "^", the class matches any character **not** in the list.

Examples:
foob[aeiou]r *finds strings 'foobar', 'foober' etc. but not 'foobbr', 'foobcr' etc.*
foob[^aeiou]r *find strings 'foobbr', 'foobcr' etc. but not 'foobar', 'foober' etc.*

Within a list, the "-" character is used to specify a **range**, so that a-z represents all characters between "a" and "z", inclusive.

If You want "-" itself to be a member of a class, put it at the start or end of the list, or escape it with a backslash. If You want ']' you may place it at the start of list or escape it with a backslash.

Examples:
[-az] *matches 'a', 'z' and '-'*
[az-] *matches 'a', 'z' and '-'*
[a\-z] *matches 'a', 'z' and '-'*
[a-z] *matches all twenty six small characters from 'a' to 'z'*
[\n-\x0D] *matches any of #10,#11,#12,#13.*
[\d-t] *matches any digit, '-' or 't'.*
[]-a] *matches any char from ']'..'a'.*

Metacharacters

Metacharacters are special characters which are the essence of Regular Expressions. There are different types of metacharacters, described below.

Metacharacters - line separators

^ *start of line*
$ *end of line*
\A *start of text*
\Z *end of text*
. *any character in line*

Examples:

^foobar *matches string 'foobar' only if it's at the beginning of line*
foobar$ *matches string 'foobar' only if it's at the end of line*
^foobar$ *matches string 'foobar' only if it's the only string in line*
foob.r *matches strings like 'foobar', 'foobbr', 'foob1r' and so on*

© 2020 LogSat Software

The "^" metacharacter by default is only guaranteed to match at the beginning of the input string/text, the "$" metacharacter only at the end. Embedded line separators will not be matched by "^'' or "$''.
You may, however, wish to treat a string as a multi-line buffer, such that the "^'' will match after any line separator within the string, and "$'' will match before any line separator. You can do this by switching On the modifier /m.
The \A and \Z are just like "^'' and "$'', except that they won't match multiple times when the modifier /m is used, while "^'' and "$'' will match at every internal line separator.

The ".'' metacharacter by default matches any character, but if You switch Off the modifier /s, then '.' won't match embedded line separators.

SpamFilter's RegEx works with line separators as recommended at www.unicode.org (http://www.unicode.org/unicode/reports/tr18/):

"^" is at the beginning of a input string, and, if modifier /m is On, also immediately following any occurrence of \x0D\x0A or \x0A or \x0D. Note that there is no empty line within the sequence \x0D\x0A.

"$" is at the end of a input string, and, if modifier /m is On, also immediately preceding any occurrence of \x0D\x0A or \x0A or \x0D. Note that there is no empty line within the sequence \x0D\x0A.

"." matches any character, but if You switch Off modifier /s then "." doesn't match \x0D \x0A and \x0A and \x0D.

Note that "^.*$" (an empty line pattern) does not match the empty string within the sequence \x0D\x0A, but matches the empty string within the sequence \x0A\x0D.

Metacharacters - predefined classes

\w	*an alphanumeric character (including "_")*
\W	*a non-alphanumeric*
\d	*a numeric character*
\D	*a non-numeric*
\s	*any space (same as [\t\n\r\f])*
\S	*a non space*

You may use \w, \d and \s within custom **character classes**.

Examples:
foob\dr *matches strings like 'foob1r', ''foob6r' and so on but not 'foobar', 'foobbr' and so on*
foob[\w\s]r *matches strings like 'foobar', 'foob r', 'foobbr' and so on but not 'foob1r', 'foob=r' and so on*

RegEx uses properties the SpaceChars and WordChars to define character classes \w, \W, \s, \S, so you can easily redefine it.

© 2020 LogSat Software

Metacharacters - word boundaries

\b *Match a word boundary*
\B *Match a non-(word boundary)*

A word boundary (\b) is a spot between two characters that has a \w on one side of it and a \W on the other side of it (in either order), counting the imaginary characters off the beginning and end of the string as matching a \W.

Metacharacters - iterators

Any item of a regular expression may be followed by another type of metacharacters - **iterators**. Using this metacharacters You can specify number of occurrences of previous character, **metacharacter** or **sub-expression**.

* *zero or more ("greedy"), similar to {0,}*
\+ *one or more ("greedy"), similar to {1,}*
? *zero or one ("greedy"), similar to {0,1}*
{n} *exactly n times ("greedy")*
{n,} *at least n times ("greedy")*
{n,m} *at least n but not more than m times ("greedy")*
*? *zero or more ("non-greedy"), similar to {0,}?*
+? *one or more ("non-greedy"), similar to {1,}?*
?? *zero or one ("non-greedy"), similar to {0,1}?*
{n}? *exactly n times ("non-greedy")*
{n,}? *at least n times ("non-greedy")*
{n,m}? *at least n but not more than m times ("non-greedy")*

So, digits in curly brackets of the form {n,m}, specify the minimum number of times to match the item n and the maximum m. The form {n} is equivalent to {n,n} and matches exactly n times. The form {n,} matches n or more times. There is no limit to the size of n or m, but large numbers will chew up more memory and slow down r.e. execution.

If a curly bracket occurs in any other context, it is treated as a regular character.

Examples:
foob.*r *matches strings like 'foobar', 'foobalkjdflkj9r' and 'foobr'*
foob.+r *matches strings like 'foobar', 'foobalkjdflkj9r' but not 'foobr'*
foob.?r *matches strings like 'foobar', 'foobbr' and 'foobr' but not 'foobalkj9r'*
fooba{2}r *matches the string 'foobaar'*
fooba{2,}r m*atchs strings like 'foobaar', 'foobaaar', 'foobaaaar' etc.*
fooba{2,3}r *matches strings like 'foobaar', or 'foobaaar' but not 'foobaaaar'*

A little explanation about "greediness". "Greedy" takes as many as possible, "non-greedy" takes as few as possible. For example, 'b+' and 'b*' applied to string 'abbbbc' return 'bbbb', 'b+?' returns 'b', 'b*?' returns empty string, 'b{2,3}?' returns 'bb', 'b {2,3}' returns 'bbb'.

You can switch all iterators into "non-greedy" mode (see the modifier /g).

© 2020 LogSat Software

Metacharacters - alternatives

You can specify a series of **alternatives** for a pattern using "|'' to separate them, so that fee|fie|foe will match any of "fee'', "fie'', or "foe'' in the target string (as would f(e|i|o)e). The first alternative includes everything from the last pattern delimiter ("('', "['', or the beginning of the pattern) up to the first "|'', and the last alternative contains everything from the last "|'' to the next pattern delimiter. For this reason, it's common practice to include alternatives in parentheses, to minimize confusion about where they start and end.
Alternatives are tried from left to right, so the first alternative found for which the entire expression matches, is the one that is chosen. This means that alternatives are not necessarily greedy. For example: when matching foo|foot against "barefoot'', only the "foo'' part will match, as that is the first alternative tried, and it successfully matches the target string. (This might not seem important, but it is important when you are capturing matched text using parentheses.)
Also remember that "|'' is interpreted as a literal within square brackets, so if You write [fee|fie|foe] You're really only matching [feio|].

Examples:
foo(bar|foo) *matches strings 'foobar' or 'foofoo'.*

Metacharacters - sub-expressions

The bracketing construct (...) may also be used for define r.e. sub-expressions (after parsing You can find sub-expression positions, lengths and actual values in MatchPos, MatchLen).

Sub-expressions are numbered based on the left to right order of their opening parenthesis.
First sub-expression has number '1' (whole r.e. match has number '0' -).

Examples:
(foobar){8,10} *matches strings which contain 8, 9 or 10 instances of the 'foobar'*
foob([0-9]|a+)r *matches 'foob0r', 'foob1r' , 'foobar', 'foobaar', 'foobaar' etc.*

Metacharacters - back-references

Metacharacters \1 through \9 are interpreted as back-references. \<n> matches previously matched **sub-expression** #<n>.

Examples:
(.)\1+ *matches 'aaaa' and 'cc'.*
(.+)\1+ *also match 'abab' and '123123'*
(['"]?)(\d+)\1 matches '"13" (in double quotes), or '4' (in single quotes) or 77 (without quotes) etc

Modifiers

© 2020 LogSat Software

There are many ways to set up modifiers.
Any of these modifiers may be embedded within the regular expression itself using the (?...) construct.

i

Do case-insensitive pattern matching (using installed in you system locale settings), see also InvertCase.

m

Treat string as multiple lines. That is, change "^'' and "$'' from matching at only the very start or end of the string to the start or end of any line anywhere within the string, see also Line separators.

s

Treat string as single line. That is, change ".'' to match any character whatsoever, even a line separators (see also Line separators), which it normally would not match.

g

Non standard modifier. Switching it Off
You'll switch all following operators into non-greedy mode (by default this modifier is On). So, if modifier /g is Off then '+' works as '+?', '*' as '*?' and so on

x

Extend your pattern's legibility by permitting whitespace and comments (see explanation below).

r

Non-standard modifier. If is set then range à-ÿ additional include russian letter '¸', À-ß additional include '¨', and à-ß include all russian symbols.
Sorry for foreign users, but it's set by default.

The modifier /x itself needs a little more explanation. It tells the TRegExpr to ignore whitespace that is neither backslashed nor within a character class. You can use this to break up your regular expression into (slightly) more readable parts. The # character is also treated as a metacharacter introducing a comment, for example:

(

© 2020 LogSat Software

```
(abc) # comment 1
  |   # You can use spaces to format r.e. -
TRegExpr ignores it
(efg) # comment 2
)
```

This also means that if you want real whitespace or # characters in the pattern (outside a character class, where they are unaffected by /x), that you'll either have to escape them or encode them using octal or hex escapes. Taken together, these features go a long way towards making regular expressions text more readable.

Perl extensions

(?imsxr-imsxr)
You may use it into r.e. for modifying modifiers by the fly. If this construction inlined into sub-expression, then it effects only into this sub-expression

Examples:

(?i)Saint-Petersburg	*matches 'Saint-petersburg' and 'Saint-Petersburg'*
(?i)Saint-(?-i)Petersburg	*matches 'Saint-Petersburg' but not 'Saint-petersburg'*
(?i)(Saint-)?Petersburg	*matches 'Saint-petersburg' and 'saint-petersburg'*
((?i)Saint-)?Petersburg	*matches 'saint-Petersburg', but not 'saint-petersburg'*

(?#text)
A comment, the text is ignored. Note that TRegExpr closes the comment as soon as it sees a ")", so there is no way to put a literal ")" in the comment.

© 2020 LogSat Software

15 SpamFilter.ini additional settings

```
[server settings]
;Set this to 1 if you want to disable EHLO extensions
DisableEHLO=0

;Any emails whose text portion exceeds this number of KB will not be scanned for keywords and Bayes
;Higher values *may* catch more spam but will cause higher load on processor
MaxMsgSizeForKeywordScan=64

;Any emails whose size exceeds this number of KB will be whitelisted. Most spam emails are small in size, lowering this value may help in reducing the chances of incorrectly blocking valid emails with large attachments. The maximum value that can be specified for this setting is 2147483648 (equal to about 2TB)
MaxMsgSizeForSpamFiltering=768

;Set FilterBase64html to 1 if you want to block any emails with Content-Transfer-Encoding=base64 and Content-Type=text/html or text/plain
FilterBase64html=0

;Set RequireHELOBeforeMAILFROM to 0 if you do not want to require remote servers to issue a HELO or EHLO command before sending the email
RequireHELOBeforeMAILFROM=1

;Controls the minimum number of good and spam emails that must be received before the Bayesian filter kicks in
MinEmailsForBayesKickIn=5000

;by default SpamFilter will not allow any IP to relay thru it. Change DoNotTrustSelfByDefault to 1 if you want localhost to be able to relay
DoNotTrustSelfByDefault=0

;Remove any stale token in the corpus db.dat file that did not appear in incoming emails for the past n days
CleanUpCorpusIntervalDays=7

;Force disconnect of sessions if a command has not been received within the last nn seconds
ReadTimeout=60

;Timeout when delivering emails to the destination SMTP server (in seconds)
ReadTimeoutOutgoing=60

;if turned on, this will cause tokens in incoming emails being logged to screen with relevant probabilities
ShowBayesianTokens=0

;Set TagSPAMAndQuarantine=1 if you want to prefix every quarantine subject line with the prefix specified in SPAMTagPrefix ini parameter
TagSPAMAndQuarantine=0
```

© 2020 LogSat Software

;This SPAMTagPrefix will be prefixed to all subject lines marked for "mark as SPAM and deliver" along with the ation specified by TagSPAMAndQuarantine
SPAMTagPrefix=SPAM:

;This SPAMTagHeader will be added to the email headers for emails marked for "mark as SPAM and deliver" along with the ation specified by TagSPAMAndQuarantine
SPAMTagHeader=X-SF-SPAM:Y

;This WhitelistTagPrefix will be inserted in the Subject line of whitelisted emails if the option to tag whitelisted emails in the subject line has been enabled
WhitelistTagPrefix=[WHITELISTED]

;The following entries in the welcome banner, the "Received:" header, and the "X-Server" headers can be customized
ResponseWelcomeBanner=Welcome to SpamFilterISP SMTP Server %Ver%
XServerHeader=LogSat Software SMTP Server
XServerBanner=LogSat Software SMTP Server

;Setting DoNotSendNDROnQuarantine to 1 will prevent generation of NDR when email are quarantined by causing SpamFilter *not* to send an error code when quarantining emails
DoNotSendNDROnQuarantine=0

;Setting IncludeOriginalMessageOnNDR to 1 will cause the entire original email to be included when sending an NDR (non-deliverable receipt). Setting this value to 0 will cause only the original email's headers to be included in the NDR
IncludeOriginalMessageOnNDR=1

;If turned on, the threads that save to disk and load into memory the bayes corpus tokens will have increased priority
BoostBayesPriority=1

;if TrailingSQLSemiColon is set to 1 SpamFilter will add a ";" to the end of SQL statements. Disable only to help solve problems with some databases.
TrailingSQLSemiColon=1

;If turned on, any quarantined (false positives) emails that the end user force-delivers will cause the sender to be automatically whitelisted (NOTE - Starting from SpamFilter v3.5, this setting has been moved to the Filters.ini file located in the appropriate directory under the \SpamFilter\Domains directory structure)
AutoWhiteListForceDeliveryEnabled=1

;if EnableBadMailDir is set to 1, this will cause all emails that generate a "server error" when forwarded to your destination SMTP server will be saved in a "BadMailDir" for troubleshooting
EnableBadMailDir=0

;if ScanReceivedHeaders is set to 1 SpamFilter will add the "Received:" headers to the text examined for keywords and statistical Bayesian searches.
ScanReceivedHeaders=1

;if ScanAllHeaders is set to 1 SpamFilter will add all email's headers to the text examined for keywords and statistical Bayesian searches.

© 2020 LogSat Software

ScanAllHeaders=0

;To avoid backscatter, if an incoming email passes all filtering rules, but cannot be forwarded (ex. mailbox full, non-existent user), SpamFilter maintains open the incoming remote connection until it can verify with the destination server that the email can be delivered. If not, a 5xx error is output forcing the remote server to generate the NDR, rather than having SpamFilter send an NDR notification email
VerifyRCPTTOForCleanEmails=1

;Path to logfile directory
LogFilePath=

;Path to queue directory
QueuePath=

;Optional destination SMTP server where to forward SPAM emails only. Good emails are still forwarded to main SMTP server
DestSMTPServerForSPAM=

;Optional destination SMTP server's port where to forward SPAM emails only. Good emails are still forwarded to main SMTP server
DestSMTPServerForSPAMPort=

;If SpamFilter is configured to "tag and deliver" emails, you can also re-route all spam emails to a different email address.
;Spam emails can either be forwarded to an "absolute" email address, or to a specific email address or each domain.
;For example, to forward *ALL* spam emails to one specific email address (i.e. "SpamBucket@mydomain.com), use the following parameter:
;
;ForwardAllSPAMtoEmailAddress=SpamBucket@mydomain.com
;
;To forward all spam emails to a specific mailbox, depending on the recipient's domain - for example to forward all spam sent to
;the mydomain1.com to "SpamBucket@mydomain1.com", and to forward all spam sent to
;the mydomain2.com to "SpamBucket@mydomain2.com",
;use the following parameter, but leaving the domain name blank. SpamFilter will dynamically append the correct email domain depending on the recipient of the spam:
;ForwardAllSPAMtoEmailAddress=SpamBucket@

;If you want to comply with RFC5321/RFC2821 and accept emails to <postmaster> without domain name, specify here a destination email where you want those emails forwarded to.
;This will allow SpamFilter to receive emails addressed to <postmaster>, without applying any filtering rules as required by the RFCs, which would otherwise be rejected.
ForwardAllPostmasterEmailsTo=

;The frequency in seconds for which the quarantine table is scanned to check for emails pending delivery - includes web-access password registration emails
QuarantineToDeliverCheckInterval=5

;By default the activity logfile is saved to disk every 60 seconds. Set

© 2020 LogSat Software

RealtimeDiskLogging=1 to save the log every time it is updated
RealtimeDiskLogging=0

;Add any IPs (separated by commas - no wildcards) that you do not wish to be automatically added to the Honeypot IP blacklist. This setting also prevents those IPs to be added to the IP cache blacklist, and to allow them to bypass the greylist filter
DoNotAddIPToHoneypot=

;An alternate server for sending NDR (non-delivery) notification emails can be used. Leave the "NotificationSMTPServer" value blank to use the default destination SMTP server
NotificationSMTPServer=
NotificationSMTPServerPort=25

;Set EnableDbgLogs=1 to enable separate detailed logging for troubleshooting purposes
EnableDbgLogs=

;The timeout in milliseconds for all DNS-related queries.
DNSTimeout=5000

;If an IP sends more than this number of spams in a certain period of time then it is temporarily banned (blacklisted)
IPCacheLimboCountTrigger=3

;If an IP sends more than a certain number of spams during this number of minutes then it is temporarily banned (blacklisted)
IPCacheLimboTimeTrigger=10

;If an IP address was banned because it sent too many spams in a certain time interval, it will be un-banned after this number of minutes
IPCacheBlacklistDuration=60

;Optionally flush to disk in the \SpamFilter\Domains directory 3 files with the contents of the list of IPs currently in memory for the IP Cache Blacklist, Limbo and GreyList. Set the interval in minutes for how often to export these lists, or enter 0 to disable.
ExportIPCachesToDiskIntervalMinutes=0

;You can force the antivirus plugin to block emails if they contain password protected archives that cannot be tested for viruses by setting this to 1
BlockArchivesWithPassword=0

;By default SpamFilter will only perform DNS lookups when the reverse DNS filter is enabled. Change value to 1 to always perform a reverse lookup on connecting IPs
AlwaysDoReverseDNSLookups=0

;Specifies how often the logfiles are rotated (Min=1, Max=24). The default is 24 (rotates at midnight). A value of 1 means every hour at the hour, value of 2 means at 2am, 4am, 6am etc...
RotateLogsEveryNNhours=24

;Change DoNotStartWithoutAV to 1 if you do not want SpamFilter to start/run if there is an error with the Antivirus plugin.
DoNotStartWithoutAV=0

© 2020 LogSat Software

;Number of hours SpamFilter will retry to deliver messages in queue to your destination SMTP server if it was unreachable. Enter 0 to try forever until back online.
ExpireRetryQueueHours=0

;Determines if SpamFilter should hold in the queue emails that were rejected by the destination SMTP server with an error in the 4xy range
QueueIfDestinationError4xy=1

;Determines if SpamFilter should hold in the queue emails that were rejected by the destination SMTP server with an error in the 5xy range
QueueIfDestinationError5xy=0

;Determines if SpamFilter should remove from the queue emails that could not be delivered to the destination SMTP server due to a "Read Timeout" (an NDR is sent if the email is removed from the queue)
DoQueueIfReadTimeout=1

;Image filter threshold. Higher values indicate a more aggressive filter. 0 disables the filter. Min=0, Max=15
SpamImageThreshold=10

;Image filter color sensitivity. Used internally to detect color shades
SpamImageColorSensitivity=20

;Images embedded in email's html having a width smaller than this will not be scanned. Useful to bypass signatures and logos
SpamImageMinWidth=300

;Images embedded in email's html having a height smaller than this will not be scanned. Useful to bypass signatures and logos
SpamImageMinHeight=300

;Determines the number of points that will be scanned in a image to process it for spam
SpamImageSamplingPoints=200

;to reduce false positives, emails with multiple inline images can bypass the image filter by setting this value to 1
SpamImagePassMultiImage=1

;Specify the max number of pages a PDF document must contain in order to be scanned for spam signatures. The scan will be skipped altogether if there are more than this number of pages. Specify 0 to disable scanning in PDF files
SpamPDFMaxPagesToScan=0

;Specify the max number height in pixel of a PDF pages that will be scanned for spam signatures. To reduce false positives, pages taller than this will not be scanned
SpamPDFMaxPixelHeight=800

;SpamFilter can block emails that contain only an empty, blank body and one of the following attachment. Clear the list if you don't want to stop such emails. Specify multiple attachments separated by commas

© 2020 LogSat Software

BlockBlankEmailsWithAttachments=*.pdf

;SpamFilter is able to block blank emails that contain specific attachments. This parameter is used to specify the threshold of characters below which an email is considered blank
MaxLettersToConsiderEmailBlank=2

;Set this to 0 to disable support for TLS
EnableTLSSupport=1

;Set this to 1 in order to disable support for the older TLS v1 protocol and only allow TLS v1.1 and TLS v1.2
DisableTLSv1_0=0

;Use this to enable/disable support for the non-secure SSLv3 to support older SMTP servers if needed
DisableSSLv3=1

;If TLS is enabled (EnableTLSSupport=1), you can enable TLS for inbound connections to SpamFilter, but not use TLS/SSL when forwarding to the destination SMTP servers by setting DisableTLSOutbound=1
DisableTLSOutbound=0

;Customize the SSL Cipher list used by SpamFilter. The list uses the OpenSSL standard syntax, and by default is AES:ALL:!aNULL:!eNULL:+RC4:@STRENGTH
SSLCipherList=

;If the private key of the SSL certificate is protected by a password, enter is here
SSLCertificatePassword=

;Some older email clients have a bug that requires them to see "AUTH=LOGIN" in the EHLO response rather than "AUTH LOGIN". Set this to 1 to add the incorrect syntax to the EHLO output. Changes to this setting require SpamFilter to be restarted
AddIncorrectAUTHLOGINEHLOEntry=0

;SpamFilter will add to the logfile the username and password used for an unsuccessful AUTH LOGIN attempt to help determine intentional password guessing attempts or user error. Change this parameter to 0 to prevent the password to be written to the logfile.
LogInvalidPasswords=1

;Timeout in seconds used in the some SQL commands (Ex. inserting a new record in the tblQuarantine table)
MiscSQLTimeout=5

;Timeout in milliseconds indicating how long SpamFilter waits for the remote destination SMTP server to acknowledge the RCPT TO command to verify the validy of the recipient for a clean email being delivered
RCPTTOVerificationTimeout=30000

;SpamFilter Enterprise will delete temporary entries in the tblReloadTableInfo after they have been kept for this long. This parameter is used to allow multiple installations of SpamFilter Enterprise to maintain their settings in sync. It can be reduced to 5-10

© 2020 LogSat Software

seconds for installations running only one instance of SpamFilter Enterprise
SecondsToHoldEntriesIntblReloadTableInfo=600

;For SpamFilter Enterprise only, the max amount of time that the process that checks for settings updates in the database is allowed to run for. Normally this takes a few tenths of a second, but if hosting tens of thousands of domains, each with customized settings, this could take longer
ChecktblReloadTableMaxRunTimeMinutes=10

;How often (in seconds) to run the process that checks connectivity on the default SMTP port and kills the entire SpamFilter process in that case so the Windows service recovery will take effect and relaunch it within a minute
WatchdogTimer=60

;If the "AuthorizedTO" whitelist is used to specify the list of valid email addresses that can be accepted, by default SpamFilter will terminate a connection when the remote server specifies an invalid address in the RCPT TO command. You can use the following option to disable this forced disconnect, and cause SpamFilter to simply reject the invalid recipient, and continue to accept additional ones
DisconnectOnNonAuthorizedTO=1

;Use this option to prevent SpamFilter from performing the routine cleanup of the quarantine database by deleting old archived emails. Useful if admins want to perform their own cleanup
DoNotDeleteExpiredEmailsFromQuarantine=0

;The amount of time in minutes the hash signature for an email content is kept in memory. If another signature is not received during this time, the has is removed from memory. Used in the SFDC filter
HashCacheBlacklistDuration=60

;The interval in seconds that must pass from the initial connection of an IP address to when it is allowed to connect to SpamFilter again. Used in the greylist filter
GreyListInterval=300

;The number of hours for which an IP will be held in the GreyList limbo before being removed. If the IP does not make a second attempt to deliver email before this many hours after the initial attempt, it will be removed from the GreyList limbo, and it will have to repeat the process next time it connects.
GreyListLimboHold=12

;The number of days for which an IP that passed the GreyList limbo and has been allowed to connect will remain allowed to do so. After this many days from the initial contact, the GreyList process will have to be repeated.
GreyListAllowedHold=60

;Set thie value to 1 to enable the GreyList filter
GreyListEnabled=0

;Set this value to 1 if you want to prevent SpamFilter from adding the "X-SF-WhiteListedReason" header in whitelisted emails
HideXSFWhiteListedReasonHeader=0

© 2020 LogSat Software

;Set this value to 1 if you want to prevent SpamFilter from logging the specific keywords that caused a reject in the "X-Rejection-Reason" header in blocked emails
HideXSFBlockedKeywordsReasonHeader=0

;If enabled SpamFilter will add an X-AuthUser header with the authenticated username for SMTP AUTH
AddHeaderForAuthUsers=1

;If AddHeaderForAuthUsers, this string will be used to identify the header with the username being added
XAuthUserString=X-SF-AuthUser

;Use this value to limit the number of nested include directives allowed in an SPF query. Used to limit the risk of DoS attacks using malicious SPF DNS records
MaxSPFAllowedLoops=20

;If using the option (disabled by default) to also test the email in the "From:" header in addition to the MAIL FROM against all blacklists/whitelistss/SPF, you can optionally choose to skip the SPF test of that header (the SPF standard only recommends to test the MAIL FROM, not the From: header)
SkipSPFWhenCheckingFromHeader=0

;If enabled, optionally force SpamFilter to perform an SPF check even if an email has already been detected as spam by another filter
CheckSPFEvenIfSpamDetected=0

;Limits the max number of recursions in RegEx expressions - used to prevent stack overflows and memory erros with complex RegEx
MaxRegExLimitRecursion=1000

;Limits the number of domains to perform lookups for the SF0Day filter, and the max number of email addresses to collect from an email when checking the SFDE filter. Useful to prvent overloads when filtering email chains with hundreds/thousands of recipient where everyone clicks on "Reply to all"
MaxDomains_SF0Day_MaxEmails_SFDE=200

;Use these parameters to limit how many emails users authenticated by AUTH LOGIN can send in a determined amount of time during the same SMTP session. Set these values to 0 to remove any limits
AUTHLOGINEmailsInIntervalMax=10
AUTHLOGINEmailsInIntervalMinutes=1000

;Add here any IPs that are allowed to use the XFORWARD extension to pass the original IP of the sender in case the email is being relayed by a server placed in front of SpamFilter, so that all IP-based tests can still be performed
AuthIPsForXforwardCommand=

;When disabling the generation of the AutoWhiteList by setting AutoWhiteListForceDeliveryEnabled=0, you can still allow SpamFilter to process the entries that exist in the WL_AutoWhiteListForceDelivery.txt whitelist by setting ProcessListEvenIfAutoWhiteListForceDeliveryDisabled=1. Useful if you manage your own

© 2020 LogSat Software

WL_AutoWhiteListForceDelivery
ProcessListEvenIfAutoWhiteListForceDeliveryDisabled=0

;SpamFilter will try to force Windows to reduce the "Memory Working Set" assigned to SpamFilter so as to lower the amount of RAM used by SpamFilter's process. Useful for 32bit implementations of SpamFilter that process over 1 million emails/day and host thousands of domains.
MemoryEmptyWorkingSetEveryNhours=24

;SpamFilter uses the http and https protocols to query the SFDB database and to download antivirus updates. You can specify a proxy to use for these operations the the option in the [proxy settings] section
[proxy settings]
ProxyServer=
ProxyUsername=
ProxyPassword=
ProxyPort=0
ProxyBasicAuthentication=0

[authentication settings]
;By setting this parameter to 1, SpamFilter will automatically add the Active Directory domain name prefix to the username. For example, if the AD domain is logsat.com, and the username is JohnW, SpamFilter will automatically authenticate the user as logsat.com\JohnW
ActiveDirectoryAuthPrefixDefaultDomain=0

;By setting this parameter to 1, SpamFilter will automatically add the Active Directory domain name suffix to the username. For example, if the AD domain is logsat.com, and the username is JohnW, SpamFilter will automatically authenticate the user as JohnW@logsat.com
ActiveDirectoryAuthAppendDefaultDomain=0

© 2020 LogSat Software

16 Purchase

Licensing for SpamFilter ISP and SpamFilter Enterprise is very simple and affordable. A license is required only for the live production server(s) running SpamFilter. A license allows for an unlimited number of users and an unlimited number of domains.

SpamFilter ISP retails for 600 USD per license.
SpamFilter Enterprise retails for 1,200 USD per license.

The license price includes 1 year of software maintenance. Additional years of software maintenance retail for 20% of the license price. A current software maintenance is required to download any updates, upgrades and patches. Support is provided for free by email. While a valid software maintenance is not required to obtain support, it will be required if a patch is needed to solve technical problems.

© 2020 LogSat Software

Index

- " -

"From Domain" = "To Domain" 9, 17
"Mail From" = "Mail To" 9, 17

- % -

% 17
%Domain% 32
%EMailFrom% 32
%EMailTo% 32
%IP% 32

- A -

Activation Code 53
Active Directory 22
Active Server Pages 56
AddIncorrectAUTHLOGINEHLOEntry 68
additional settings 68
Allow % in address 17
Allowed domains 13
AlwaysDoReverseDNSLookups 68
Antivirus 53
anti-virus 53
ASP 55, 56, 57
Attachment 9, 19
Attachment Blocking 10
AUTH LOGIN 22
Authentication 56
Authorized TO Emails 13, 22
AutoWhiteListForceDeliveryEnabled 68

- B -

Bayesian 10, 30
Blacklisted Domains 9, 19
Blacklisted FROM Emails 9, 19
Blacklisted IPs 9, 19
Blacklisted TO Emails 9, 19
BlockArchivesWithPassword 68
BoostBayesPriority 68

- C -

CA 58, 59
Cached IP Blocking 15
Cert.pem 58, 59
certificate 58, 59
Certificate Authority 58, 59
cleanup interval 35
CleanUpCorpusIntervalDays 68
console 49
Copy file-based settings 51
Country Blacklist 10
Country Filters 9, 19

- D -

DameWare 49
database 38, 40, 49
Database Configuration 38
Database Setup 35
Databases 38
db_connect.asp 56
DBAs 35
DestSMTPServerForSPAM 68
DisableEHLO 68
distribution 7
DNA 30
DNS 10, 15, 27
DNSTimeout 68
Domain Blacklist 10
DoNotAddIPToHoneypot 68
DoNotQueueIfReadTimeout 68
DoNotSendNDROnQuarantine 68
DoNotStartWithoutAV 68
DoNotTrustSelfByDefault 68
DSN 42

- E -

Empty "Mail From" 9, 17
EnableBadMailDir 68
EnableDbgLogs 68
Enterprise 51
Enterprise Mode 38
Excluded Domains 13
Excluded Domains / IPs 22

© 2020 LogSat Software

Excluded FROM Emails 13, 22
Excluded IPs 13
ExpireRetryQueueHours 68

- F -

Fail 27
feature comparison 9
FilterBase64html 68
Filters.ini 51
FQDN 7, 15
FROM EMail Blacklist 10
Fully Qualified Domain Name 15

- G -

GeoIP 10
GUI 49

- H -

Honeypot 9, 10, 19

- I -

IdleDisconnectMinutesTimeout 68
IIS 56
Image filter 9, 10, 17
Import settings 51
IP Blacklist 10
IP caching 10
IPCacheBlacklistDuration 68
IPCacheLimboCountTrigger 68
IPCacheLimboTimeTrigger 68

- K -

Key.pem 58, 59
Keyword 9, 19
Keyword Content Filtering 10
Keyword whitelisting 13
Keywords 9, 19, 22

- L -

LDAP 22
Licensing 77
local domain 13
Local Domains 22
log files 55
LogFilePath 68
Logging 15

- M -

MAPS 9, 10, 17, 19
MAPS Servers 10
Max concurrent incoming SMTP connections 15
Max Email Size 15
Max Recipients 15
MaxMind 10
MaxMsgSizeForKeywordScan 68
metacharacter 61
Microsoft Access 35, 38
Microsoft IIS 56
Microsoft Outlook 55, 57
Microsoft SQL Server 35, 38, 40
MinEmailsForBayesKickIn 68
MiscSQLTimeout 68
mstsc 49
multiple 49
multiple servers 49
MX 9, 10, 17, 44
MySQL 35, 38, 42

- N -

Neutral 27
NoNDR 19
Norman 53
NotificationSMTPServer 68
NotificationSMTPServerPort 68
NULL 19

- O -

ODBC 42
ODBC Data Sources 42
offline 38
OLE DB 40, 42
OpenSSL 58, 59
Oracle 35, 38
Outlook 55, 57

© 2020 LogSat Software

- P -

parsed 55
Pass 27
Password 22, 56
patch 77
PCAnywhere 49
Personal Exchange Format 59
pfx 59
PHP 55, 56
plug-in 55
Port 15
Prerequisites 38
private key 58
Process queue 15
proxy 68
Proxy Basic Authentication 68
Proxy Password 68
Proxy Port 68
Proxy Server 68
Proxy Username 68
PTR 10

- Q -

quarantined 35
QuarantineToDeliverCheckInterval 68
QueueIfDestinationError400 68
QueueIfDestinationError500 68
Quick Setup 7

- R -

RCPT TO 22
RCPT TOs 10
RDP 49
ReadTimeout 68
ReadTimeoutOutgoing 68
RealtimeDiskLogging 68
RegEx 19, 22, 61
regular expression 24
Regular Expressions 19, 22, 61
Remote Desktop 49
RequireHELOBeforeMAILFROM 68
reverse DNS 9, 10, 17
Root.pem 58, 59
RotateLogsEveryNNhours 68

- S -

Save Settings 7
Sawmill 55
ScanAllHeaders 68
ScanReceivedHeaders 68
Sender Policy Framework 10
service 7, 49
SFDB 25
SFDB Filter 10
ShowBayesianTokens 68
SMTP listener socket 15
SMTP Settings 7
SMTP User Authentication 22
Softfail 27
software maintenance 77
spaces 9, 17
SpamFilter Distributed Blacklist 25
SpamFilter Enterprise 51
SpamFilter.exe 7, 49
SpamFilter.ini 51, 68
SpamFilterMSSQL.udl 40
SpamFilterMySQL.udl 42
SpamFilterSvc.exe 49
SpamImageColorSensitivity 68
SpamImageMinHeight 68
SpamImageMinWidth 68
SpamImagePassMultiImage 68
SpamImageSamplingPoints 68
SpamImageThreshold 68
SPAMTagPrefix 68
SPF 10, 27
SpoolQueueFilesToMemory 68
SQL Server 35, 38, 40
SSL 15, 58, 59
SSL Port 15
SSLCertificatePassword 68
Standalone 7
Standard 38
Standard Mode 38
subject line 9, 17
Supported Databases 35
SURBL 9, 10, 19
Syntax 61

© 2020 LogSat Software

- T -

table sizes 35
TAG 17, 22
TagSPAMAndQuarantine 68
TAGSUBJECT 22
tblLogins 35, 56
tblMsgs 35
tblQuarantine 35
tblRejectCodes 35
tblServers 35
Terminal Services 49
threshold 9, 17
TO EMail Blacklist 10
TrailingSQLSemiColon 68
triggers 38

- U -

UDL 42, 56
UDL files 35
Unfiltered Emails 13, 22
Unix 22
updates 77
upgrade 51
upgrades 77
Upgrading SpamFilter ISP Standard to SpamFilter Enterprise 51
User Authentication 13, 22, 56
Username 56

- V -

VNC 49

- W -

Web interface 55, 56, 57
Windows service 49
wizard 35, 40

- X -

x.509 58
XServerHeader 68

- Z -

ZIP 7

© 2020 LogSat Software